Обробка зображень і мультимедіа

Олег Гутік

Лекція 12: Стиснення зображень, VI. Ортогональні перетворення

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Стиснення зображень. Ортогональні перетворення

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Стиснення зображень. Ортогональні перетворення

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Стиснення зображень. Ортогональні перетворення

Перетворення, які використовуються для стиснення зображень, повинні бути швидкими і, по можливості, легко реалізованими на комп'ютері. Це передусім передбачає, такі перетворення мають бути *лінійними*. Тобто, перетворені величини c_i є лінійними комбінаціями (сумами з деякими множниками або вагами) вихідних величин (пікселів) d_j, причому відповідним множником або вагою є деяке число w_{ij} (коефіцієнт перетворення). Отже,

$$c_i = \sum_j d_j w_{ij},$$

де $i, j = 1, 2, \ldots, n$. Наприклад, у випадку $n = 4$ це перетворення можна записати в матричній формі

$$\begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} w_{11} & w_{12} & w_{13} & w_{14} \\ w_{21} & w_{22} & w_{23} & w_{24} \\ w_{31} & w_{32} & w_{33} & w_{34} \\ w_{41} & w_{42} & w_{43} & w_{44} \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{pmatrix},$$

яка в загальному випадку набуде такого вигляду: $C = W \cdot D$. Кожен вектор-стовпець матриці W називається "*базисним вектором*".

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \dots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Стиснення зображень. Ортогональні перетворення

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

Важливим завданням є визначення коефіцієнтів перетворення w_{ij}. Основна вимога полягає в тому, щоб після перетворення величина c_1 була б більшою, а решта величин $c_2, c_3, \ldots, c_n$ стали б малими. Основне співвідношення $c_i = \sum_j d_j w_{ij}$ передбачає, що c_i буде більшим, якщо ваги w_{ij} посилюватимуть відповідні величини d_j. Це станеться, наприклад, якщо компоненти векторів w_{ij} та d_j мають близькі значення та однакові знаки. Навпаки, величина c_i буде малою, якщо ваги w_{ij} будуть малими, і половина з них матиме знак, протилежний знаку відповідного числа d_j. Тому, якщо виходять великі c_i, то вектори w_{ij} мають подібність з вихідним вектором d_j, а малі величини c_i означають, що компоненти w_{ij} сильно відрізняються від d_j. Отже, базисні вектори w_{ij} можна інтерпретувати як інструмент добування деяких характерних ознак вихідного вектора.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

Стиснення зображень. Ортогональні перетворення

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

На практиці ваги w_{ij} не повинні залежати від вихідних даних. Інакше їх доведеться додавати в стислий файл для використання декодером. Це міркування, і навіть те що, що вихідні дані є пікселами, тобто, невід'ємними величинами, визначає спосіб вибору базисних векторів. Перший вектор, той, який, породжує величину c_i, має складатися з близьких, можливо, чисел, що збігаються. Він посилюватиме невід'ємні величини пікселів. А решта векторів базису мають наполовину складатися з додатних чисел, а на іншу половину — з від'ємних. Після множення на додатні величини та їхнє додавання результат буде малим числом. (Це особливо вірно, коли вихідні дані близькі, а ми знаємо, що сусідні пікселі мають, зазвичай, близькі величини.) Нагадаємо, що базисні вектори є деяким інструментом для отримання особливостей вихідних даних. Тому хорошим вибором будуть базисні вектори, які сильно відрізняються один від одного і тому можуть отримувати різні особливості. Це призводить до думки, що базисні вектори повинні бути взаємно ортогональні. Якщо матриця перетворення W складається з ортогональних векторів, то перетворення називається *ортогональним*. Інше спостереження, що дозволяє правильно вибирати базисні вектори, полягає в тому, що ці вектори повинні мати все більші частоти зміни знаку, щоб видобувати, так би мовити, високочастотні характеристики даних, що стискаються при обчисленні перетворених величин.

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \qquad (1)$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \qquad (1)$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \qquad (1)$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \tag{1}$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \tag{1}$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \tag{1}$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \quad (1)$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \tag{1}$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \quad (1)$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Цим властивостям задовольняє така ортогональна матриця:

$$W = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}. \qquad (1)$$

Перший базовий вектор (верхній рядок матриці W) складається з одних одиниць, а тому його частота дорівнює нулю. Всі інші вектори мають дві $+1$ і дві -1, а тому вони дадуть малі перетворені величини, які частоти (виміряні кількістю змін символів у рядку) зростають. Ця матриця подібна до матриці перетворення Адамара-Волша. Наприклад, перетворимо початковий вектор $(4, 6, 5, 2)$:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} 4 \\ 6 \\ 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 17 \\ 3 \\ -5 \\ 1 \end{pmatrix}.$$

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Результат цілком оптимістичний, оскільки число c_i стало більшим (порівняно з початковими даними), а два інших числа стали меншими. Обчислимо енергії вихідних та перетворених даних. Початкова енергія дорівнює

$$4^2 + 6^2 + 5^2 + 2^2 = 81,$$

а після перетворення енергія стала

$$17^2 + 3^2 + (-5)^2 + 1^2 = 324,$$

що вчетверо більше. Енергію можна зберегти, якщо помножити матрицю перетворення W на коефіцієнт $1/2$. Новий добуток $W \cdot (4, 6, 5, 2)^T$ дорівнюватиме

$$(17/2, 3/2, -5/2, 1/2).$$

Отже, енергія зберігається та концентрується в першій компоненті, і вона тепер складає $8.5^2/81 = 89\%$ від загальної енергії початкових даних, в яких на частку першої компоненти припадало всього 20%.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4,6,5,2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4,6,5,2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4,6,5,2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Інша перевага матриці W полягає в тому, що вона ж робить обернене перетворення. Вихідні дані $(4, 6, 5, 2)$ відновлюються за допомогою добутку

$$W \cdot (17/2, 3/2, -5/2, 1/2)^T.$$

Тепер ми можемо оцінити переваги цього перетворення. Квантуємо перетворений вектор $(8.5, 1.5, -2.5, 0.5)$ за допомогою його округлення до цілого та отримуємо $(9, 1, -3, 0)$. Робимо обернене перетворення та отримуємо вектор $(3.5, 6.5, 5.5, 2.5)$. В аналогічному експерименті ми просто вилучимо два найменші числа та отримаємо $(8.5, 0, -2.5, 0)$, а потім зробимо обернене перетворення цього грубо квантованого вектора. Це призводить до відновлених даних $(3, 5.5, 5.5, 3)$, які дуже близькі до вихідних. Отже, наш висновок: *навіть це просте та інтуїтивне перетворення є гарним інструментом для "вичавлювання" надмірності з вихідних даних.* Більш витончені перетворення дають результати, які дозволяють відновлювати дані з високим степенем схожості навіть при досить грубому квантуванні.

Дякую за увагу!