Stationarity

DS-GA 1013 / MATH-GA 2824 Mathematical Tools for Data Science

https://cims.nyu.edu/~cfgranda/pages/MTDS_spring20/index.html

Carlos Fernandez-Granda

Motivation

Goal: Estimate signal $y \in \mathbb{R}^N$ from noisy data $x \in \mathbb{R}^N$

Regression problem

Optimal estimator?

Linear estimator?

Translation

Linear translation-invariant models

Stationary signals and PCA

Wiener filtering

Circular translation

We focus on circular translations that wrap around

We denote by $x^{\downarrow s}$ the sth circular translation of a vector $x \in \mathbb{C}^N$

For all $0 \leq j \leq N-1$,

$$x^{\downarrow s}[j] = x[(j-s) \bmod N]$$

Circular translation in 2D

For an $N \times N$ signal $X \in \mathbb{C}^{N \times N}$, circular translation by $(s_1, s_2) \in$ is denoted by $X^{\downarrow(s_1,s_2)}$

For all $0 \leq j_1, j_2 \leq N-1$,

$$X^{\downarrow(s_1,s_2)}[j_1, j_2] = X[(j_1 - s_1) \bmod N, (j_2 - s_2) \bmod N].$$

Effect of shift on sinusoids

Shifting a sinusoid modifies its phase

$$\begin{aligned}\psi_k^{\downarrow s}[l] &= \exp\left(\frac{i2\pi k(l-s)}{N}\right)\\ &= \exp\left(-\frac{i2\pi ks}{N}\right)\psi_k[l]\end{aligned}$$

Effect of shift on sinusoids

Shifting a sinusoid modifies its phase

$$\begin{aligned}\Phi_{k_1,k_2}^{\downarrow(s_1,s_2)} &= \psi_{k_1}^{\downarrow s_1}(\psi_{k_2}^{\downarrow s_2})^T \\ &= \exp\left(-\frac{i2\pi k_1 s_1}{N}\right)\exp\left(-\frac{i2\pi k_2 s_2}{N}\right)\Phi_{k_1,k_2}\end{aligned}$$

Effect of translation in Fourier domain

Let $x \in \mathbb{C}^N$ with DFT $\hat{x}$ and $y := x^{\downarrow s}$

$$
\begin{aligned}
\hat{y}[k] &:= \langle x^{\downarrow s}, \psi_k \rangle \\
&= \langle x, \psi_k^{\downarrow -s} \rangle \\
&= \left\langle x, \exp\left(\frac{i2\pi ks}{N}\right) \psi_k \right\rangle \\
&= \exp\left(-\frac{i2\pi ks}{N}\right) \langle x, \psi_k \rangle \\
&= {\color{red}\exp\left(-\frac{i2\pi ks}{N}\right)} \hat{x}[k]
\end{aligned}
$$

Effect of translation in Fourier domain

Let $X \in \mathbb{C}^{N \times N}$ with DFT $\widehat{X}$ and $Y := X^{\downarrow (s_1, s_2)}$, $0 \leq s_1, s_2 \leq N-1$

$$\widehat{Y}[k_1, k_2] := \exp\left(-\frac{i2\pi k s_1}{N}\right) \exp\left(-\frac{i2\pi k s_2}{N}\right) \widehat{X}[k_1, k_2], \quad 1 \leq k_1, k_2 \leq N$$

Translation

Linear translation-invariant models

Stationary signals and PCA

Wiener filtering

Linear translation-invariant (LTI) function

A function $\mathcal{F}$ from $\mathbb{C}^N$ to $\mathbb{C}^N$ is linear if for any $x, y \in \mathbb{C}^N$ and any $\alpha \in \mathbb{C}$

$$\mathcal{F}(x + y) = \mathcal{F}(x) + \mathcal{F}(y),$$
$$\mathcal{F}(\alpha x) = \alpha \mathcal{F}(x),$$

and translation invariant if for any shift $0 \leq s \leq N - 1$

$$\mathcal{F}(x^{\downarrow s}) = \mathcal{F}(x)^{\downarrow s}$$

Linear translation-invariant (LTI) function

A function $\mathcal{F}$ from $\mathbb{C}^{N \times N}$ to $\mathbb{C}^{N \times N}$ is linear if for any $X, Y \in \mathbb{C}^{N \times N}$ and any $\alpha \in \mathbb{C}$

$$\mathcal{F}(X + Y) = \mathcal{F}(X) + \mathcal{F}(Y),$$
$$\mathcal{F}(\alpha X) = \alpha \mathcal{F}(X),$$

and translation invariant if for any $0 \leq s_1, s_2 \leq N - 1$

$$\mathcal{F}(X^{\downarrow(s_1,s_2)}) = \mathcal{F}(X)^{\downarrow(s_1,s_2)}$$

Parametrizing a linear function

Let e_j be the jth standard vector ($e_j[j] = 1$ and $e_j[k] = 0$ for $k \neq j$)

Let $\mathcal{F}_L : \mathbb{C}^N \to \mathbb{C}^N$ be a linear function

$$\begin{aligned}
\mathcal{F}_L(x) &= \mathcal{F}_L\left(\sum_{j=0}^{N-1} x[j] e_j\right) \\
&= \sum_{j=0}^{N-1} x[j] \mathcal{F}_L(e_j) \\
&= \begin{bmatrix} \mathcal{F}_L(e_0) & \mathcal{F}_L(e_1) & \cdots & \mathcal{F}_L(e_{N-1}) \end{bmatrix} x \\
&= Mx
\end{aligned}$$

Parametrizing an LTI function

Let $\mathcal{F} : \mathbb{C}^N \to \mathbb{C}^N$ be linear and translation invariant

$$\begin{aligned}
\mathcal{F}_L(x) &= \mathcal{F}\left(\sum_{j=0}^{N-1} x[j] e_j\right) \\
&= \sum_{j=0}^{N-1} x[j]\mathcal{F}(e_j) \\
&= \sum_{j=0}^{N-1} x[j]\mathcal{F}\left(e_0^{\downarrow j}\right) \\
&= \sum_{j=0}^{N-1} x[j]\mathcal{F}(e_0)^{\downarrow j}
\end{aligned}$$

Impulse response

Standard basis vectors can be interpreted as *impulses*

LTI are characterized by their impulse response

$$h_{\mathcal{F}} := \mathcal{F}(e_0)$$

In 2D

$$H_{\mathcal{F}} := \mathcal{F}(E_0)$$

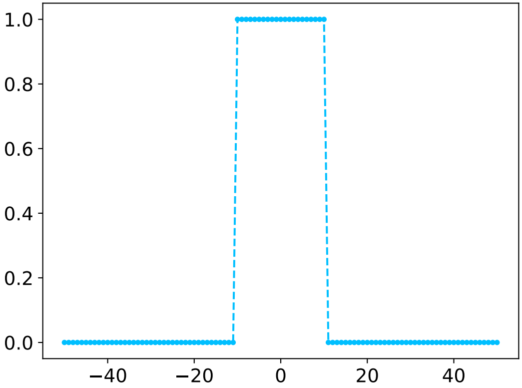
where $E[0,0] = 1$ and $E[j_1, j_2] = 0$ otherwise

Circular convolution

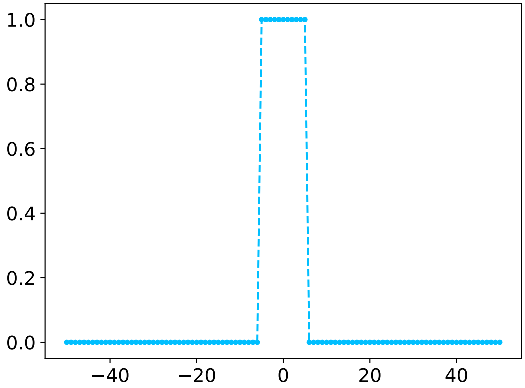
The circular convolution between two vectors $x, y \in \mathbb{C}^N$ is defined as

$$x * y\,[j] := \sum_{s=0}^{N-1} x\,[s]\, y^{\downarrow s}\,[j]\,, \quad 0 \leq j \leq N-1$$

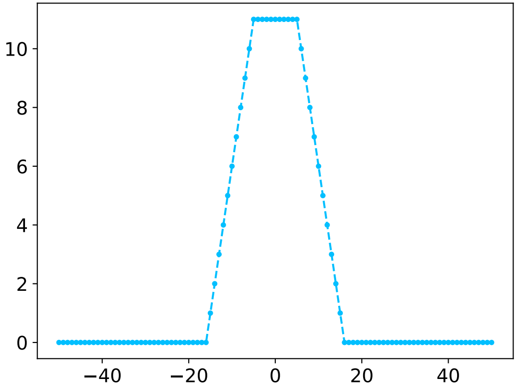
Convolution example: x



Convolution example: y



Convolution example: $x * y$

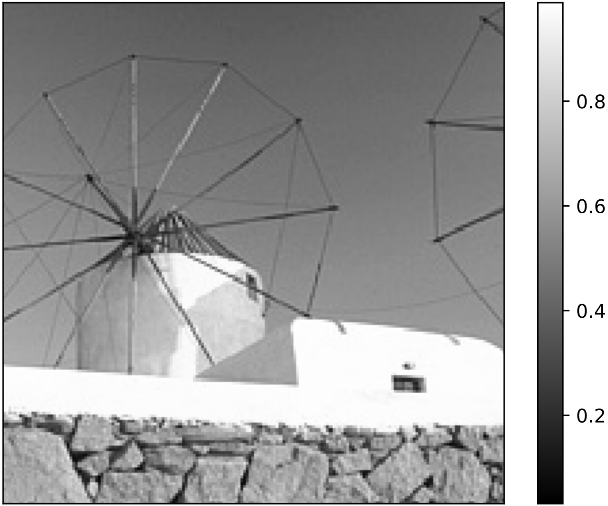


Circular convolution

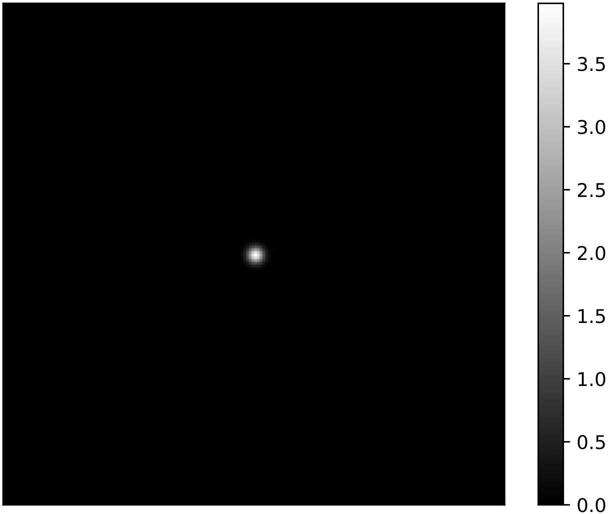
The 2D circular convolution between $X \in \mathbb{C}^{N \times N}$ and $Y \in \mathbb{C}^{N \times N}$ is

$$X * Y\left[j_1, j_2\right] := \sum_{s_1=0}^{N-1} \sum_{s_2=0}^{N-1} X\left[s_1, s_2\right] Y^{\downarrow(s_1,s_2)}\left[j_1, j_2\right], \quad 0 \leq j_1, j_2 \leq N-1$$

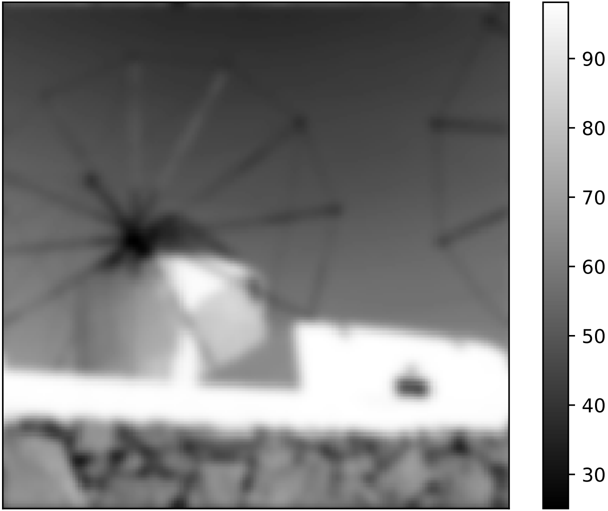
Convolution example: x



Convolution example: y



Convolution example: $x * y$



LTI functions as convolution with impulse response

For any LTI function $\mathcal{F} : \mathbb{C}^N \to \mathbb{C}^N$ and any $x \in \mathbb{C}^N$

$$\begin{aligned} \mathcal{F}(x) &= \sum_{j=0}^{N-1} x[j] \mathcal{F}(e_0)^{\downarrow j} \\ &= x * h_{\mathcal{F}} \end{aligned}$$

For any 2D LTI function $\mathcal{F} : \mathbb{C}^{N \times N} \to \mathbb{C}^{N \times N}$ and any $X \in \mathbb{C}^{N \times N}$

$$\mathcal{F}(X) = X * H_{\mathcal{F}}$$

Convolution in time is multiplication in frequency

Let $y := x_1 * x_2$, $x_1, x_2 \in \mathbb{C}^N$. Then

$$\hat{y}\,[k] = \hat{x}_1\,[k]\,\hat{x}_2\,[k]\,, \quad 0 \leq k \leq N-1$$

Convolution in time is multiplication in frequency

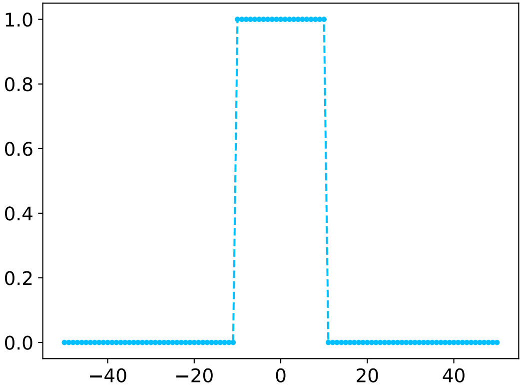
Let $Y := X_1 * X_2$ for $X_1, X_2 \in \mathbb{C}^{N \times N}$. Then

$$\widehat{Y}[k_1, k_2] = \widehat{X_1}[k_1, k_2] \, \widehat{X_2}[k_1, k_2]$$

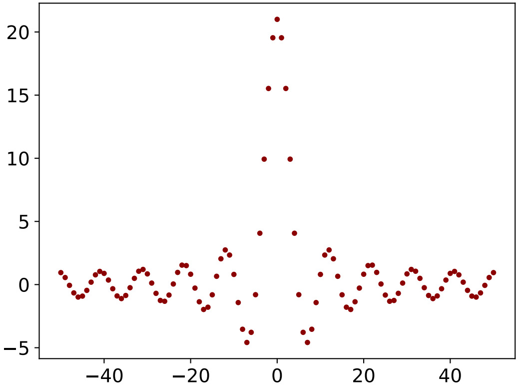
Proof

$$
\begin{aligned}
\hat{y}\,[k] &:= \langle x_1 * x_2, \psi_k \rangle \\
&= \left\langle \sum_{s=0}^{N-1} x_1\,[s]\, x_2^{\downarrow s}, \psi_k \right\rangle \\
&= \left\langle \sum_{s=0}^{N-1} x_1\,[s]\, \frac{1}{N} \sum_{j=0}^{N-1} \exp\left(-\frac{i2\pi js}{N}\right) \hat{x}_2[j]\psi_j, \psi_k \right\rangle \\
&= \sum_{j=0}^{N-1} \hat{x}_2[j] \frac{1}{N} \langle \psi_j, \psi_k \rangle \sum_{s=0}^{N-1} x_1\,[s] \exp\left(-\frac{i2\pi js}{N}\right) \\
&= \sum_{j=0}^{N-1} \hat{x}_1[j]\hat{x}_2[j] \frac{1}{N} \langle \psi_j, \psi_k \rangle \\
&= \hat{x}_1[k]\hat{x}_2[k]
\end{aligned}
$$

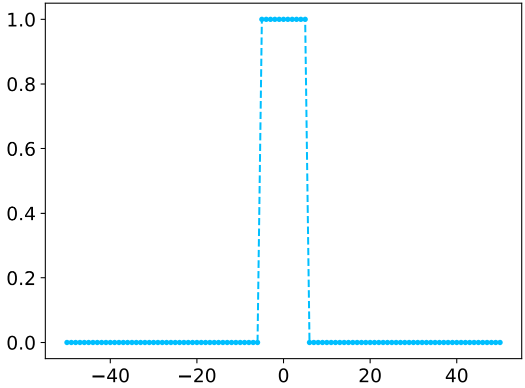
x



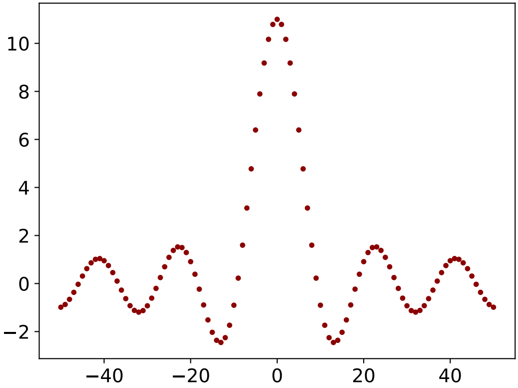
$\hat{x}$



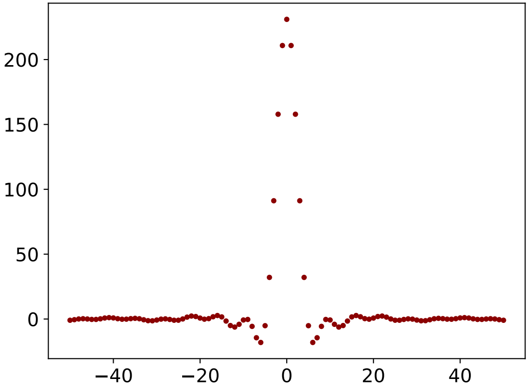
y



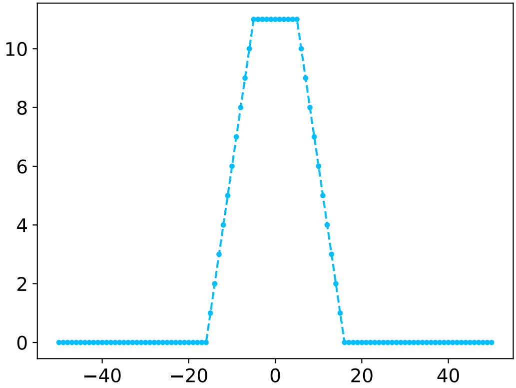
$\hat{y}$



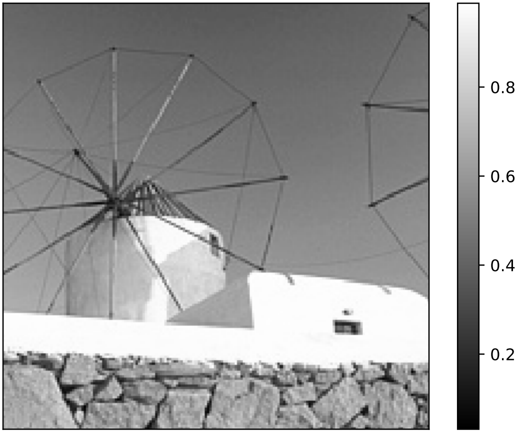
$\hat{x} \circ \hat{y}$



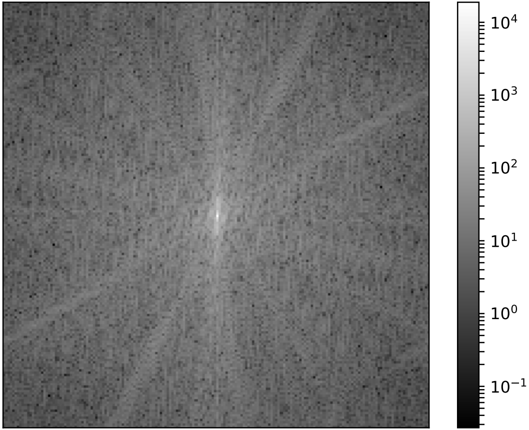
$x * y$



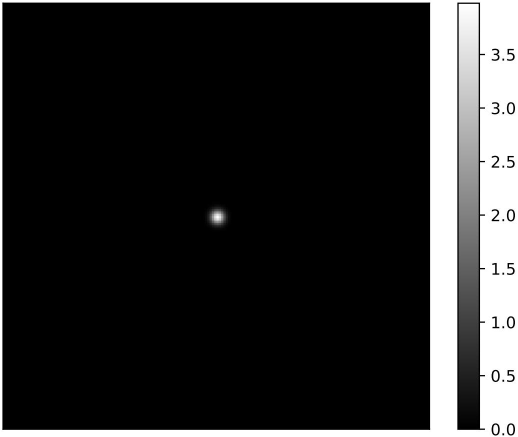
X



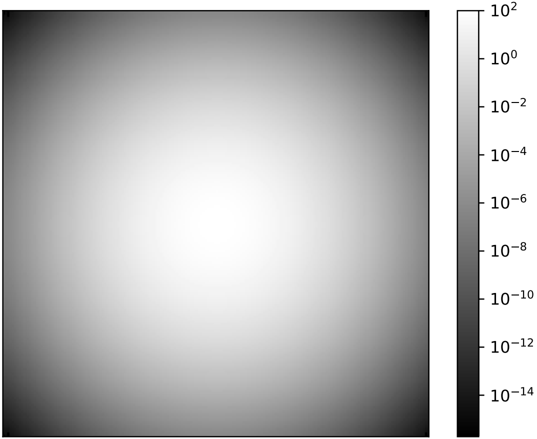
$\widehat{X}$



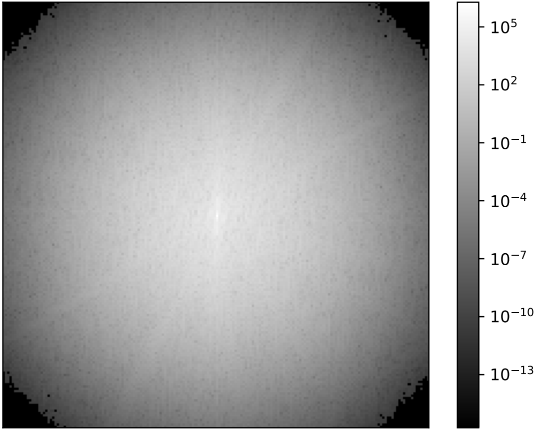
Y



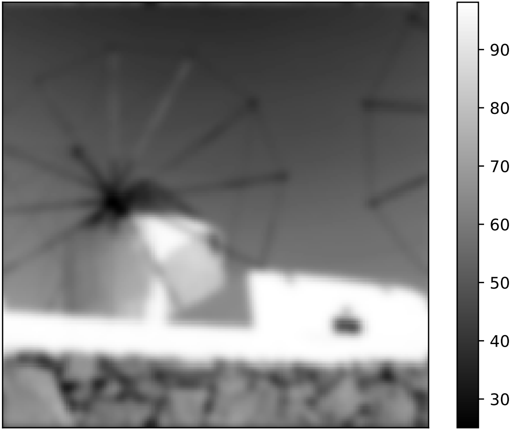
$\hat{Y}$



$\widehat{X} \circ \widehat{Y}$



$X * Y$



Convolution in time is multiplication in frequency

LTI functions just scale Fourier coefficients!

DFT of impulse response is the transfer function of the function

For any LTI function $\mathcal{F}$ and any $x \in C^N$

$$\mathcal{F}(x) = \sum_{k=0}^{N-1} \hat{h}_{\mathcal{F}}[k]\hat{x}[k]\psi_k.$$

For any 2D LTI function $\mathcal{F}$ and any $X \in C^{N \times N}$

$$\mathcal{F}(X) = \sum_{k_1=0}^{N-1} \sum_{k_2=1}^{N} \hat{H}_{\mathcal{F}}[k_1, k_2]\widehat{X}[k_1, k_2]\Phi_{k_1,k_2}$$

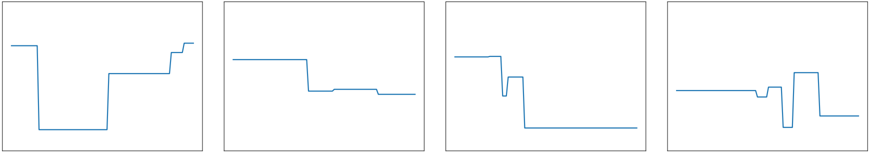
Translation

Linear translation-invariant models

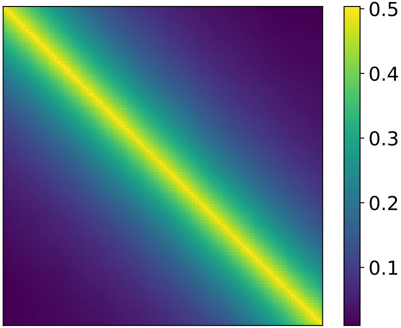
Stationary signals and PCA

Wiener filtering

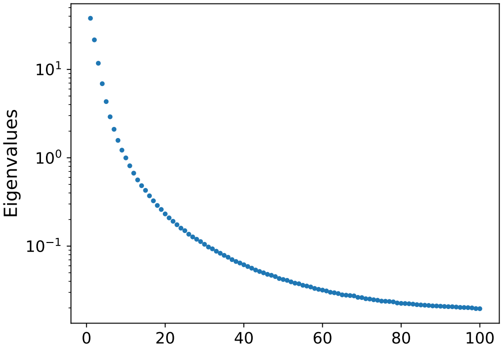
Signal with translation-invariant statistics



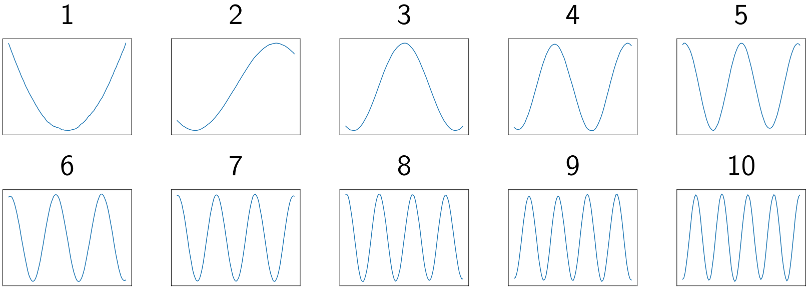
Sample covariance matrix



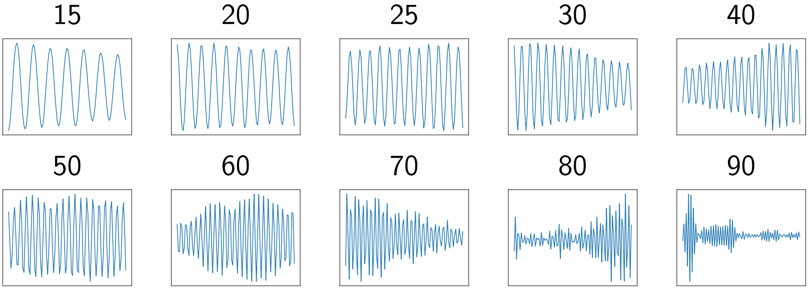
Eigenvalues



Principal directions



Principal directions



Stationary signals

$\tilde{x}$ is wide-sense or weak-sense stationary if

1. it has a constant mean

$$\mathrm{E}\left(\tilde{x}[j]\right) = \mu, \quad 1 \leq j \leq N$$

2. there is a function $a_{\tilde{x}}$ such that

$$\mathrm{E}\left(\tilde{x}[j_1]\tilde{x}[j_2]\right) = \mathrm{ac}_{\tilde{x}}(j_2 - j_1 \bmod N), \quad 0 \leq j_1, j_2 \leq N-1$$

   i.e. it has translation-invariant covariance

Autocovariance

$\mathrm{ac}_{\tilde{x}}$ is the autocovariance of $\tilde{x}$

For any j, $\mathrm{ac}_{\tilde{x}}(j) = \mathrm{ac}_{\tilde{x}}(-j) = \mathrm{ac}_{\tilde{x}}(N-j)$

$$\Sigma_{\tilde{x}} = \begin{bmatrix} \mathrm{ac}_{\tilde{x}}(0) & \mathrm{ac}_{\tilde{x}}(1) & \cdots & \mathrm{ac}_{\tilde{x}}(1) \\ \mathrm{ac}_{\tilde{x}}(1) & \mathrm{ac}_{\tilde{x}}(0) & \cdots & \mathrm{ac}_{\tilde{x}}(2) \\ & & \cdots & \\ \mathrm{ac}_{\tilde{x}}(1) & \mathrm{ac}_{\tilde{x}}(2) & \cdots & \mathrm{ac}_{\tilde{x}}(0) \end{bmatrix}$$
$$= \begin{bmatrix} a_{\tilde{x}} & a_{\tilde{x}}^{\downarrow 1} & a_{\tilde{x}}^{\downarrow 2} & \cdots a_{\tilde{x}}^{\downarrow N-1} \end{bmatrix}^T$$

where

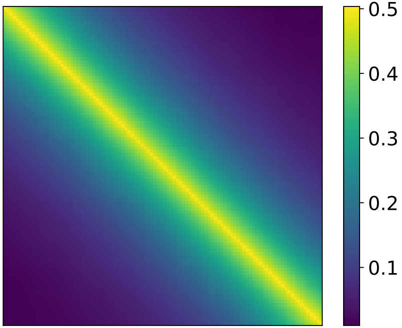
$$a_{\tilde{x}} := \begin{bmatrix} \mathrm{ac}_{\tilde{x}}(0) \\ \mathrm{ac}_{\tilde{x}}(1) \\ \mathrm{ac}_{\tilde{x}}(2) \\ \cdots \end{bmatrix}$$

Circulant matrix

Each row vector is a unit circular shift of previous row

$$\begin{bmatrix} a & b & c & d \\ d & a & b & c \\ c & d & a & b \\ b & c & d & a \end{bmatrix}$$

Sample covariance matrix



Eigendecomposition of circulant matrix

Any circulant matrix $C \in \mathbb{C}^{n \times n}$ can be written as

$$C := \frac{1}{N} F_{[N]}^{*} \Lambda F_{[N]}$$

where $F_{[N]}$ is the DFT matrix and Λ is a diagonal matrix

Proof

For any vector $x \in \mathbb{C}^n$

$$\begin{aligned} Cx &= c * x \\ &= \frac{1}{N} F_{[N]}^* \operatorname{diag}(\hat{c}) F_{[N]} x \end{aligned}$$

Eigendecomposition of circulant covariance matrix

A valid eigendecomposition is given by

$$\frac{1}{\sqrt{N}} F_{[N]}^{*} \operatorname{diag}(\hat{c}) \frac{1}{\sqrt{N}} F_{[N]}$$

If $\hat{c}$ have different values, singular vectors are sinusoids!

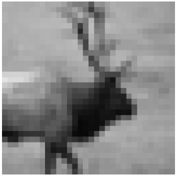
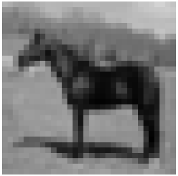
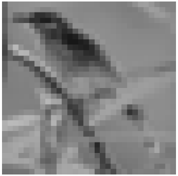
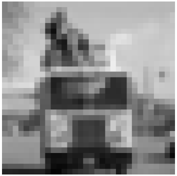
PCA on stationary vector

Let $\tilde{x}$ be wide-sense stationary with autocovariance vector $a_{\tilde{x}}$

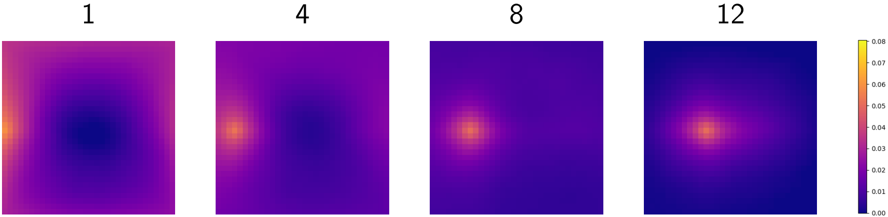
The eigendecomposition of the covariance matrix of $\tilde{x}$ equals

$$\Sigma_{\tilde{x}} = \frac{1}{N} F^{*} \operatorname{diag}(\hat{a}_{\tilde{x}}) F$$

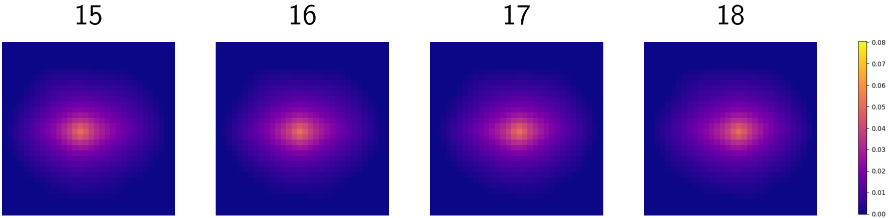
CIFAR-10 images

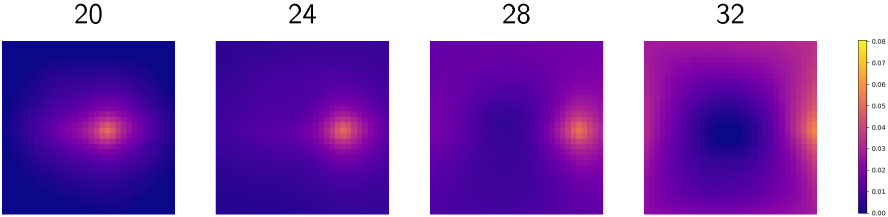
Rows of covariance matrix



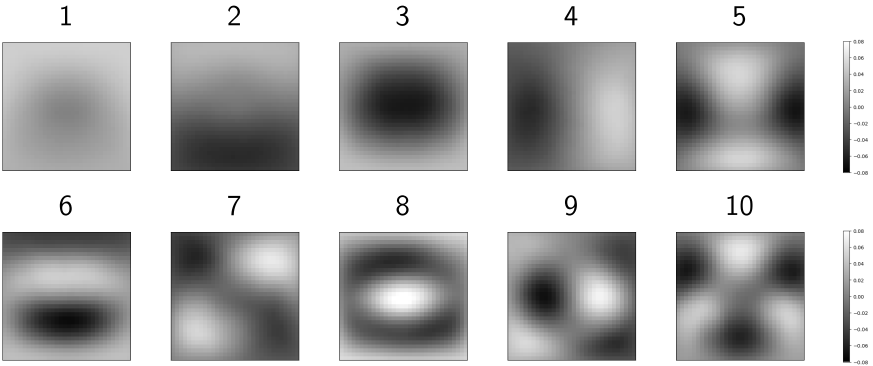
Rows of covariance matrix



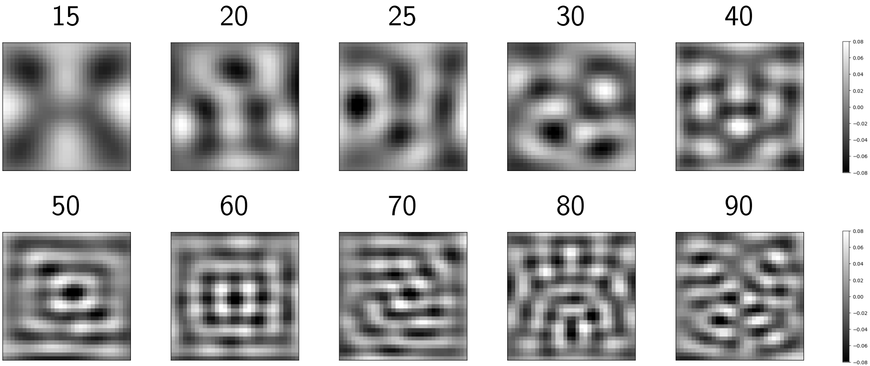
Rows of covariance matrix



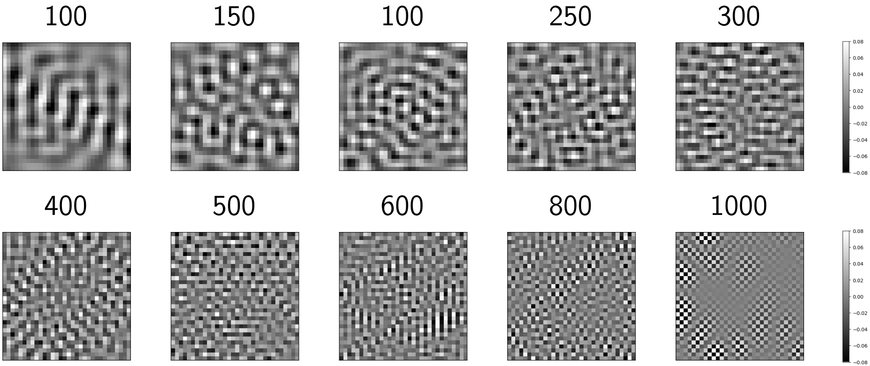
Principal directions



Principal directions



Principal directions



PCA of natural images

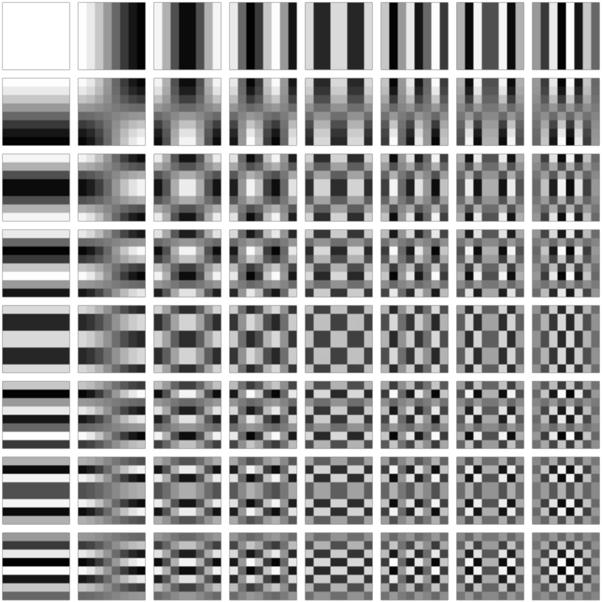
Principal directions tend to be sinusoidal

This suggests using 2D sinusoids for dimensionality reduction

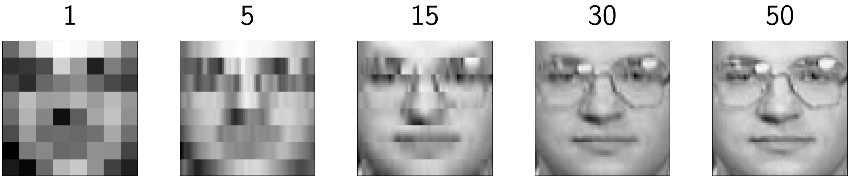
JPEG compresses images using discrete cosine transform (DCT):

1. Image is divided into 8×8 patches
2. Each DCT band is quantized differently (more bits for lower frequencies)

DCT basis vectors



Projection of each 8x8 block onto first DCT coefficients



Translation

Linear translation-invariant models

Stationary signals and PCA

Wiener filtering

Signal estimation

Goal: Estimate N-dimensional signal from N-dimensional data

Minimum MSE estimator is conditional mean (usually intractable)

Linear minimum MSE estimator?

Linear MMSE

Let $\tilde{y}$ and $\tilde{x}$ be N-dimensional zero-mean random vectors

If $\Sigma_{\tilde{x}}$ is full rank, then

$$\Sigma_{\tilde{x}}^{-1}\Sigma_{\tilde{x}\tilde{y}} := \arg\min_{B} \mathrm{E}\left(\left|\left| \tilde{y} - B^T \tilde{x} \right|\right|_2^2 \right)$$

$$\Sigma_{\tilde{x}\tilde{y}} := \mathrm{E}\left(\tilde{x}\tilde{y}^T \right)$$

Proof

The cost function can be decomposed into

$$\mathrm{E}\left(\left|\left|\tilde{y} - B^T \tilde{x}\right|\right|_2^2\right) = \sum_{j=1}^{n} \mathrm{E}\left[\left(\tilde{y}[j] - B_j^T \tilde{x}\right)^2\right]$$

Each one is a linear regression problem with optimal estimator

$$\Sigma_{\tilde{x}}^{-1} (\Sigma_{\tilde{x}\tilde{y}})_j = \arg\min_{B_j} \mathrm{E}\left[(\tilde{y}[j] - \tilde{x}^T B_j)^2\right]$$

where $(\Sigma_{\tilde{x}\tilde{y}})_j$ is the jth column of $\Sigma_{\tilde{x}\tilde{y}}$

Joint stationarity

$\tilde{x}$ and $\tilde{y}$ are jointly wide-sense or weak-sense stationary if

1. they are each wide-sense or weak-sense stationary
2. there is a function $\text{cc}_{\tilde{x},\tilde{y}}$ such that

$$\mathrm{E}\left(\tilde{x}[j_1]\tilde{y}[j_2]\right) = \text{cc}_{\tilde{x}\tilde{y}}(j_2 - j_1 \bmod N), \quad 0 \leq j_1, j_2 \leq N-1$$

   i.e. they have translation-invariant cross-covariance

Cross-covariance

$\text{cc}_{\tilde{x}\tilde{y}}$ is the cross-covariance of $\tilde{x}$ and $\tilde{y}$

$$\Sigma_{\tilde{x}\tilde{y}} = \begin{bmatrix} \text{cc}_{\tilde{x}\tilde{y}}(0) & \text{cc}_{\tilde{x}\tilde{y}}(1) & \cdots & \text{cc}_{\tilde{x}\tilde{y}}(-1) \\ \text{cc}_{\tilde{x}\tilde{y}}(-1) & \text{cc}_{\tilde{x}\tilde{y}}(0) & \cdots & \text{cc}_{\tilde{x}\tilde{y}}(2) \\ & & \cdots & \\ \text{cc}_{\tilde{x}\tilde{y}}(1) & \text{cc}_{\tilde{x}\tilde{y}}(2) & \cdots & \text{cc}_{\tilde{x}\tilde{y}}(0) \end{bmatrix}$$

$$= \begin{bmatrix} c_{\tilde{x}\tilde{y}} & c_{\tilde{x}}^{\downarrow 1} & c_{\tilde{x}}^{\downarrow 2} & \cdots c_{\tilde{x}}^{\downarrow N-1} \end{bmatrix}^T$$

where

$$c_{\tilde{x}\tilde{y}} := \begin{bmatrix} \text{cc}_{\tilde{x}\tilde{y}}(0) \\ \text{cc}_{\tilde{x}\tilde{y}}(1) \\ \text{cc}_{\tilde{x}\tilde{y}}(2) \\ \cdots \end{bmatrix}$$

Wiener filter

Let $\tilde{x}$ and $\tilde{y}$ be zero-mean and jointly stationary

The linear estimate of $\tilde{y}$ given $\tilde{x}$ that minimizes MSE as the convolution of $\tilde{x}$ with the Wiener filter w, defined by

$$\hat{w}[k] := \frac{\mathrm{Cov}(\tilde{x}_F[k], \tilde{y}_F[k])}{\mathrm{Var}(\tilde{x}_F[k])}, \quad 0 \le k \le N-1$$

where $\tilde{x}_F$ and $\tilde{y}_F$ denote the DFT coefficients of $\tilde{x}$ and $\tilde{y}$, and

$$\begin{aligned}\mathrm{Cov}(\tilde{x}_F[k], \tilde{y}_F[k]) &:= \mathrm{E}\left(\tilde{x}_F[k]\overline{\tilde{y}_F[k]}\right) \\ \mathrm{Var}(\tilde{x}_F[k]) &:= \mathrm{E}\left(|\tilde{x}_F[k]|^2\right), \quad 0 \le k \le N-1\end{aligned}$$

Proof

$$\begin{aligned}
\Sigma_{\tilde{x}} &= \frac{1}{N} F^* \operatorname{diag}(\hat{a}_{\tilde{x}}) F \\
\Sigma_{\tilde{x}\tilde{y}} &= \frac{1}{N} F^* \operatorname{diag}(\hat{c}_{\tilde{x}}) F \\
\Sigma_{\tilde{x}\tilde{y}}^T \Sigma_{\tilde{x}}^{-1} &= (\frac{1}{N} F^* \operatorname{diag}(\hat{a}_{\tilde{x}}) F)^{-1} \frac{1}{N} F^* \operatorname{diag}(\hat{c}_{\tilde{x}}) F \\
&= F^* \operatorname{diag}(\hat{a}_{\tilde{x}}^{-1}) F F^* \operatorname{diag}(\hat{c}_{\tilde{x}}) F \\
&= F^* \operatorname{diag}(\hat{a}_{\tilde{x}}^{-1} \hat{c}_{\tilde{x}}) F
\end{aligned}$$

Proof

$$
\begin{aligned}
\Sigma_{\tilde{x}_F} &:= \mathrm{E}\left(F\tilde{x}(F\tilde{x})^*\right) \\
&= F\mathrm{E}\left(\tilde{x}\tilde{x}^T\right)F^* \\
&= F\Sigma_{\tilde{x}}F^* \\
&= F\frac{1}{N}F^*\operatorname{diag}(\hat{a}_{\tilde{x}})FF^* \\
&= N\operatorname{diag}(\hat{a}_{\tilde{x}})
\end{aligned}
$$

$$
\hat{a}_{\tilde{x}}[k] = \frac{\mathrm{Var}(\tilde{x}_F[k])}{N}, \quad 0 \leq k \leq N-1
$$

Proof

$$
\begin{aligned}
\Sigma_{\tilde{x}_F \tilde{y}_F} &:= \mathrm{E}\left(F\tilde{x}(F\tilde{y})^*\right) \\
&= F\mathrm{E}\left(\tilde{x}\tilde{y}^T\right) F^* \\
&= F\Sigma_{\tilde{x}\tilde{y}} F^* \\
&= F\frac{1}{N}F^* \operatorname{diag}(\hat{c}_{\tilde{x}})FF^* \\
&= N \operatorname{diag}(\hat{c}_{\tilde{x}})
\end{aligned}
$$

$$
\hat{c}_{\tilde{x}\tilde{y}}[k] = \frac{\mathrm{Cov}(\tilde{x}_F[k], \tilde{y}_F[k])}{N}, \quad 0 \le k \le N-1
$$

Proof

$$\begin{aligned}
\Sigma_{\tilde{x}\tilde{y}}^{T}\Sigma_{\tilde{x}}^{-1} &= F^{*}\,\mathrm{diag}(\hat{a}_{\tilde{x}}^{-1}\hat{c}_{\tilde{x}})F \\
&= F^{*}\,\mathrm{diag}_{k=0}^{N-1}\left(\frac{\mathrm{Cov}(\tilde{x}_F[k],\tilde{y}_F[k])}{\mathrm{Var}(\tilde{x}_F[k])}\right)F
\end{aligned}$$

Least squares

Training set (y_1, x_1), (y_2, x_2), ..., (y_n, x_n) optimal LTI estimator

$$w := \arg\min_{v} \sum_{j=1}^{n} ||y_j - v * x_j||_2^2$$

is Wiener filter with transfer function

$$\hat{w} = \frac{\operatorname{cov}\left(\hat{\mathcal{X}}[k], \hat{\mathcal{Y}}[k]\right)}{\operatorname{var}\left(\hat{\mathcal{X}}[k]\right)}, \quad 0 \leq k \leq N-1$$

where

$$\operatorname{cov}\left(\hat{\mathcal{X}}[k], \hat{\mathcal{Y}}[k]\right) := \frac{1}{n} \sum_{j=1}^{n} \hat{x}_j[k] \overline{\hat{y}_j[k]}$$

$$\operatorname{var}\left(\hat{\mathcal{X}}[k]\right) := \frac{1}{n} \sum_{j=1}^{n} |\hat{x}_j[k]|^2, \quad 0 \leq k \leq N-1$$

Proof

$$\begin{aligned}\sum_{j=1}^{n} ||y_j - v * x_j||_2^2 &= \sum_{j=1}^{n} \left|\left|\frac{1}{N} F^*(\hat{y}_j - \hat{v} \circ \hat{x}_j)\right|\right|_2^2 \\ &= \frac{1}{N^2} \sum_{j=1}^{n} ||\hat{y}_j - \hat{v} \circ \hat{x}_j||_2^2 \\ &= \frac{1}{N^2} \sum_{j=1}^{n} \sum_{k=1}^{N} |\hat{y}_j[k] - \hat{v}[k]\hat{x}_j[k]|^2 := \frac{1}{N^2} \sum_{k=1}^{N} C_k\left(\hat{v}[k]\right)\end{aligned}$$

Denoising

Measurements

$$\tilde{x} = \tilde{y} + \tilde{z},$$

where $\tilde{z}$ is zero-mean Gaussian noise with variance σ^2, independent of $\tilde{y}$

Noise

Linear transformation $A\tilde{z}$ of a Gaussian vector with mean $\vec{\mu}$ and covariance matrix Σ is Gaussian with mean $A\vec{\mu}$ and cov. matrix $A\Sigma A^*$

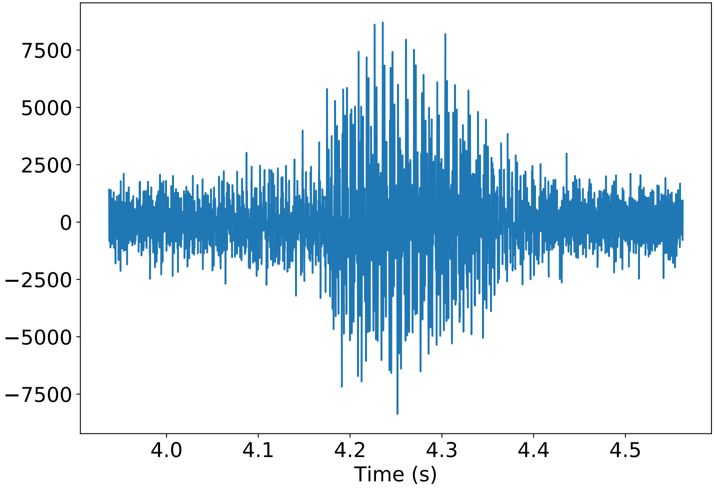
Fourier coefficients of noise are Gaussian with zero mean and covariance matrix $F_{[N]}\sigma^2 I F_{[N]}^* = N\sigma^2 I$ (iid Gaussian with variance $N\sigma^2$)

Wiener filter

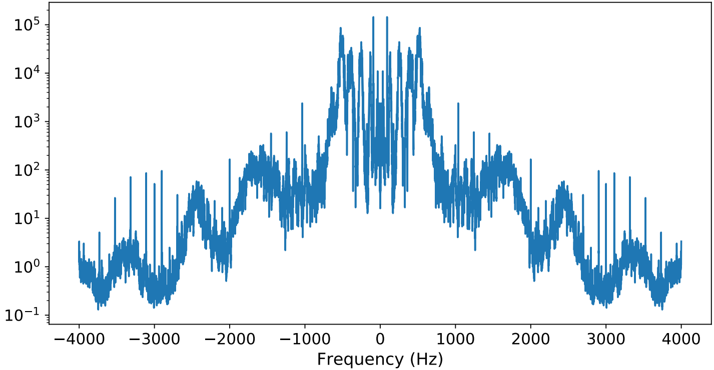
$$\begin{aligned}
\mathrm{Cov}(\tilde{x}_F[k], \tilde{y}_F[k]) &= \mathrm{E}\left(\tilde{x}_F[k]\overline{\tilde{y}_F[k]}\right) && (1)\\
&= \mathrm{E}\left(\tilde{y}_F[k]\overline{\tilde{y}_F[k]}\right) + \mathrm{E}\left(\tilde{z}_F[k]\overline{\tilde{y}_F[k]}\right) && (2)\\
&= \mathrm{Var}\left(\tilde{y}_F[k]\right) && (3)\\
\mathrm{Var}(\tilde{x}_F[k]) &= \mathrm{Var}\left(\tilde{y}_F[k]\right) + \mathrm{Var}\left(\tilde{z}_F[k]\right) && (4)\\
&= \mathrm{Var}\left(\tilde{y}_F[k]\right) + \sigma^2 && (5)\\
& && (6)
\end{aligned}$$

$$\hat{w}[k] = \frac{\mathrm{Var}\left(\tilde{y}_F[k]\right)}{\mathrm{Var}\left(\tilde{y}_F[k]\right) + \sigma^2}, \quad 0 \leq k \leq N-1 \tag{7}$$

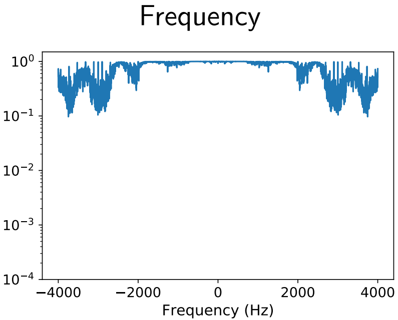
Audio data

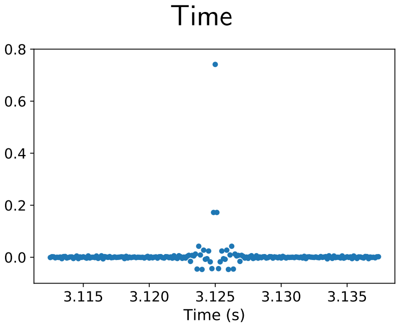


Audio data: Variance of Fourier coefficients

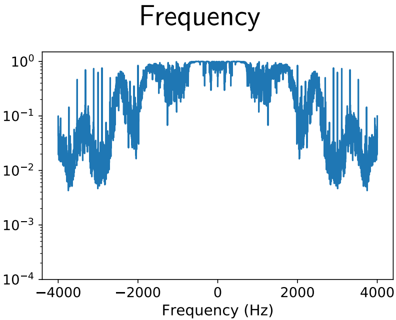


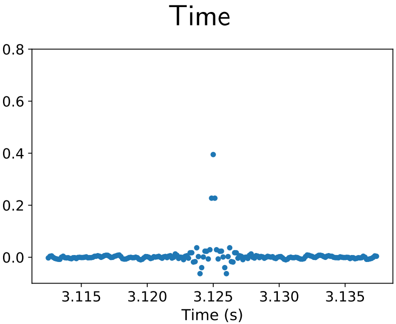
Wiener filter: $\sigma = 0.02$



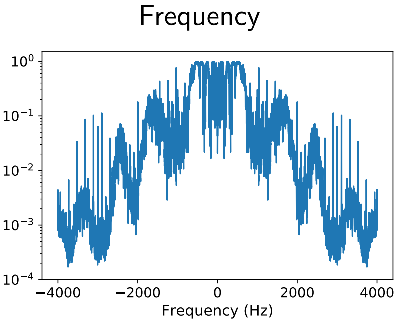


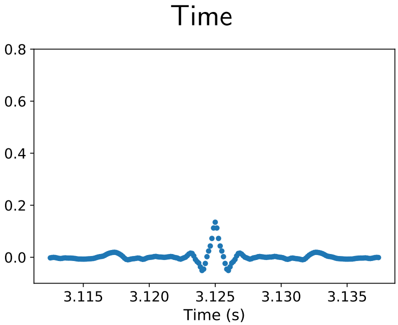
Wiener filter: $\sigma = 0.1$



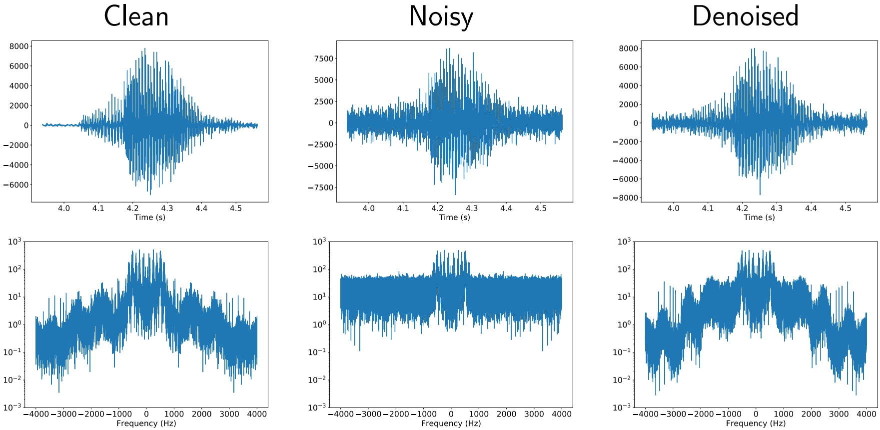


Wiener filter: $\sigma = 0.5$

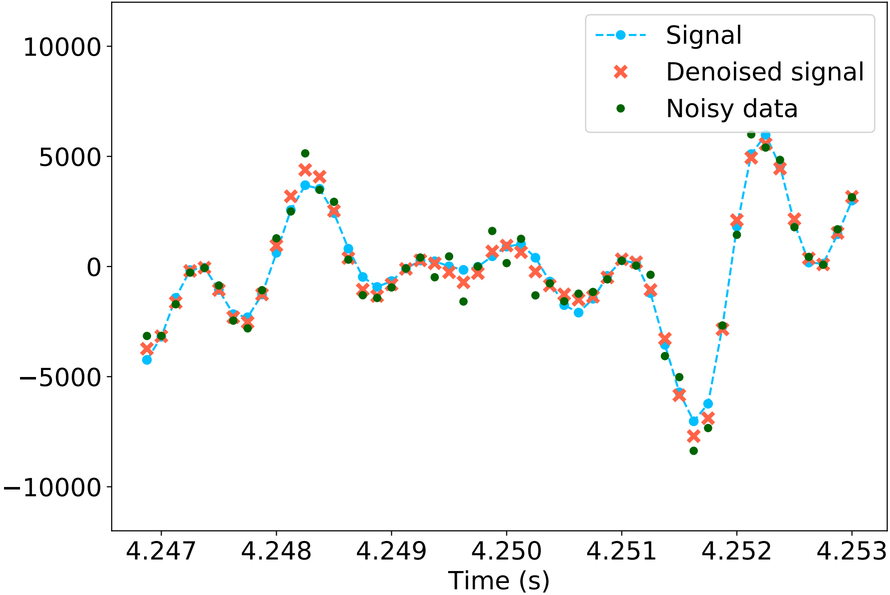




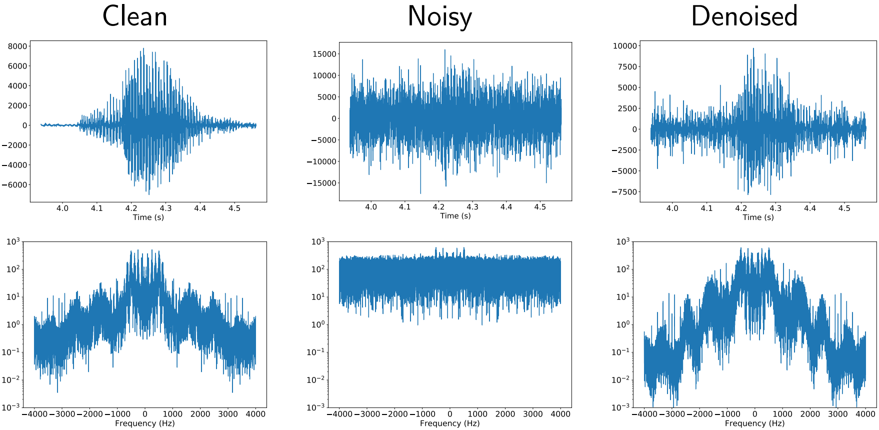
Example: $\sigma = 0.1$



Example: $\sigma = 0.1$



Example: $\sigma = 0.5$



Example: $\sigma = 0.5$

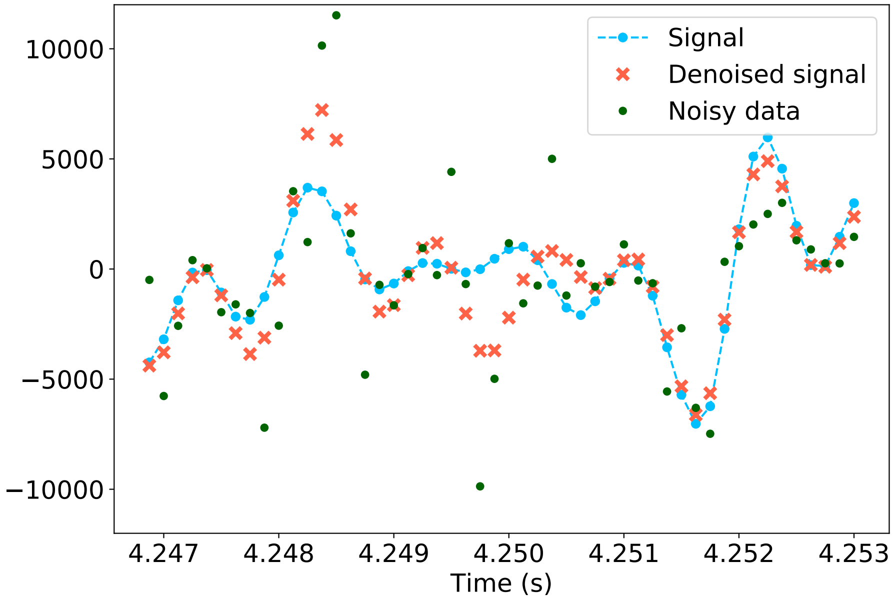


Image data

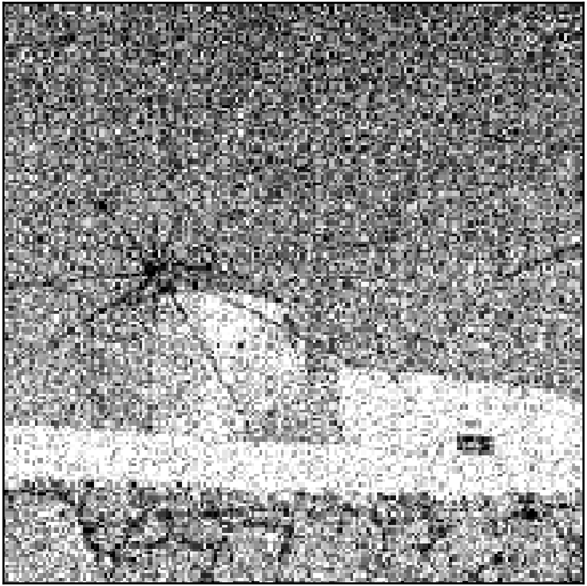
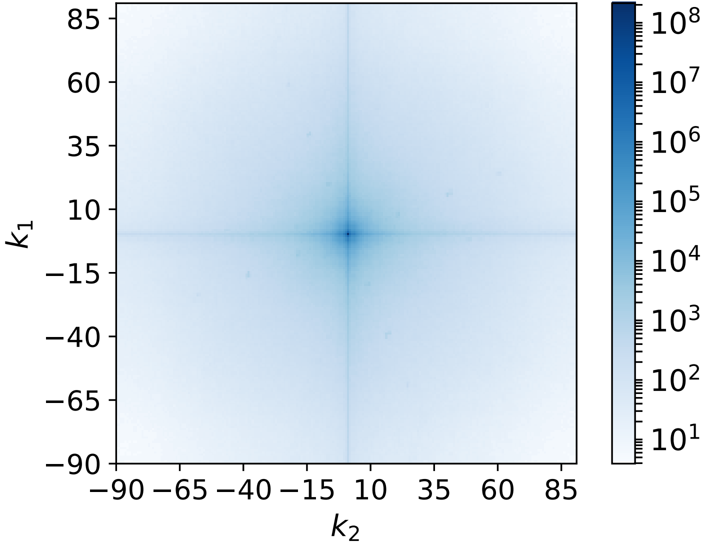
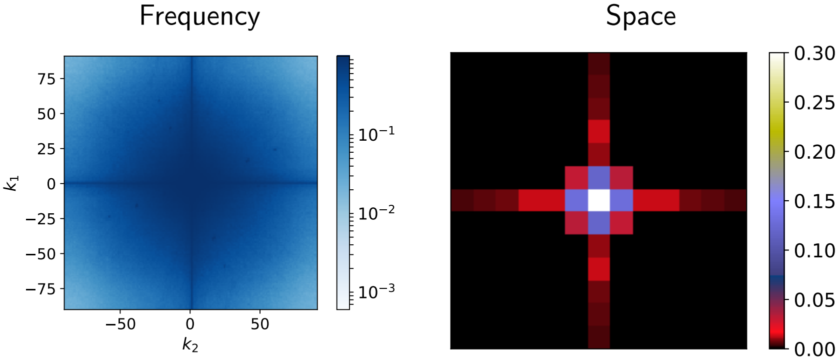


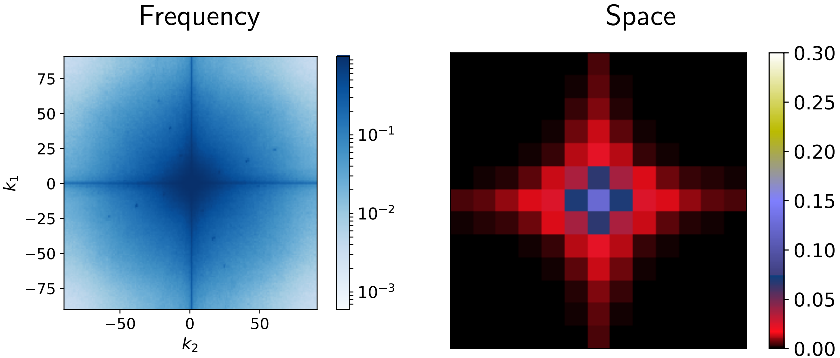
Image data: Variance of Fourier coefficients



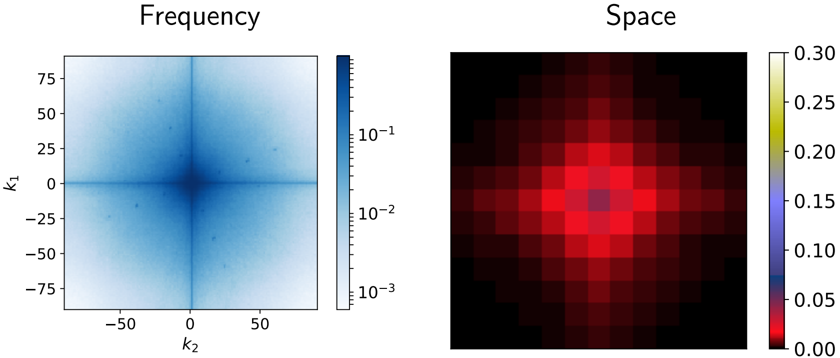
Wiener filter: $\sigma = 0.04$



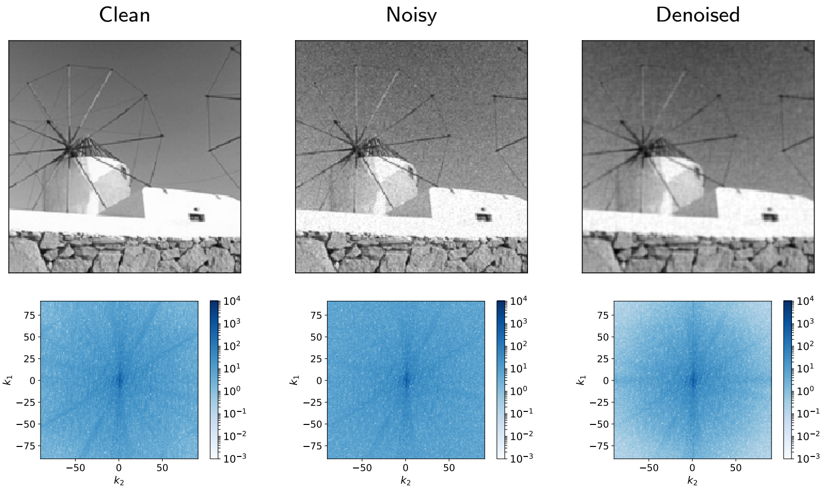
Wiener filter: $\sigma = 0.1$



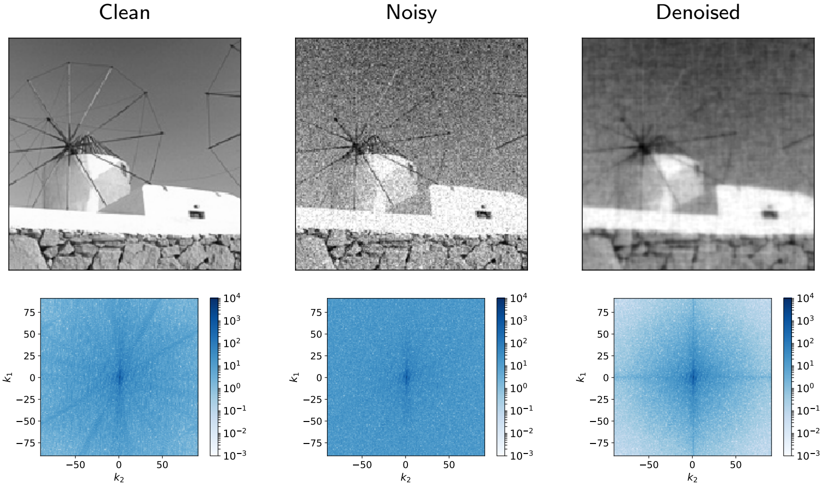
Wiener filter: $\sigma = 0.2$



Example: $\sigma = 0.04$



Example: $\sigma = 0.1$



Example: $\sigma = 0.2$