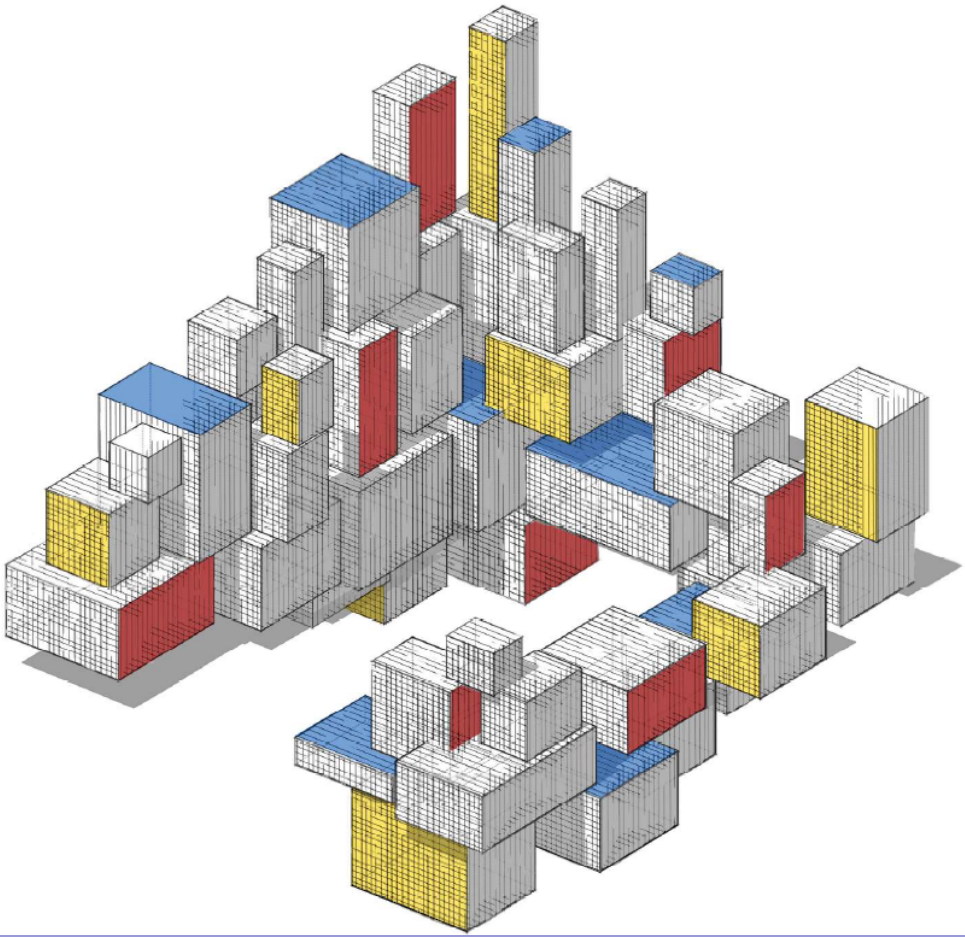
Inference for Normal Linear Models



Purpose

In this lecture we discuss:

Inference for Normal Linear Models

So far we have not assumed any distribution for the random vector of errors $\boldsymbol{\varepsilon} = [\varepsilon_1, \ldots, \varepsilon_n]^\top$ in a linear model $\boldsymbol{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$.

When the error terms $\{\varepsilon_i\}$ are assumed to be normally distributed (that is, $\{\varepsilon_i\} \sim_{\text{iid}} \mathcal{N}(0, \sigma^2)$), whole new avenues open up for inference on linear models.

We already saw that for such normal linear models, estimation of $\boldsymbol{\beta}$ and σ^2 can be carried out via maximum likelihood methods, yielding the estimators

- $\widehat{\boldsymbol{\beta}} = \mathbf{X}^+ \boldsymbol{y}$
- $\widehat{\sigma^2} = \|\boldsymbol{y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|^2 / n$

The next theorem shows that $\widehat{\boldsymbol{\beta}}$ and $\widehat{\sigma^2} n/(n-p)$ are independent and unbiased estimators of $\boldsymbol{\beta}$ and σ^2, respectively.

Theorem: Properties of the Estimators for a Normal Linear Model

Consider the linear model $\boldsymbol{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, with $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_n)$, where $\boldsymbol{\beta}$ is a p-dimensional vector of parameters and σ^2 a dispersion parameter. The following results hold.

1. The maximum likelihood estimators $\widehat{\boldsymbol{\beta}}$ and $\widehat{\sigma^2}$ are independent.
2. $\widehat{\boldsymbol{\beta}} \sim \mathcal{N}(\boldsymbol{\beta},\ \sigma^2 (\mathbf{X}^\top \mathbf{X})^+)$.
3. $n\,\widehat{\sigma^2}/\sigma^2 \sim \chi^2_{n-p}$, where $p = \text{rank}(\mathbf{X})$.

Proof

Since $\widehat{\boldsymbol{\beta}} = \mathbf{X}^{+}\boldsymbol{Y}$ is a linear transformation of a normal random vector, it has a multivariate normal distribution.

The mean vector and covariance matrix follow from:

$$\mathbb{E}\,\widehat{\boldsymbol{\beta}} = \mathbf{X}^{+}\,\mathbb{E}\boldsymbol{Y} = \mathbf{X}^{+}\mathbf{X}\,\boldsymbol{\beta} = \boldsymbol{\beta}$$

and

$$\mathbb{C}\mathrm{ov}(\widehat{\boldsymbol{\beta}}) = \mathbf{X}^{+}\sigma^{2}\mathbf{I}_{n}(\mathbf{X}^{+})^{\top} = \sigma^{2}(\mathbf{X}^{\top}\mathbf{X})^{+}.$$

Theorem: Normal and Noncentral χ^2 Distributions

Let $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{I}_n)$ be an n-dimensional normal random vector and let $\mathcal{V}_k \subset \mathcal{V}_m$ be linear subspaces of dimensions k and m, respectively, with $k < m \leqslant n$. Let $\boldsymbol{X}_k$ and $\boldsymbol{X}_m$ be orthogonal projections of $\boldsymbol{X}$ onto $\mathcal{V}_k$ and $\mathcal{V}_m$, and let $\boldsymbol{\mu}_k$ and $\boldsymbol{\mu}_m$ be the corresponding projections of $\boldsymbol{\mu}$. Then, the following holds.

1. The random vectors $\boldsymbol{X}_k$, $\boldsymbol{X}_m - \boldsymbol{X}_k$, and $\boldsymbol{X} - \boldsymbol{X}_m$ are independent.
2. $\|\boldsymbol{X}_k\|^2 \sim \chi^2_k(\|\boldsymbol{\mu}_k\|)$, $\|\boldsymbol{X}_m - \boldsymbol{X}_k\|^2 \sim \chi^2_{m-k}(\|\boldsymbol{\mu}_m - \boldsymbol{\mu}_k\|)$, and $\|\boldsymbol{X} - \boldsymbol{X}_m\|^2 \sim \chi^2_{n-m}(\|\boldsymbol{\mu} - \boldsymbol{\mu}_m\|)$.

Proof

Define $\boldsymbol{Y}^{(2)} = \mathbf{X}\widehat{\boldsymbol{\beta}}$. Note that $\boldsymbol{Y}/\sigma$ has a $\mathcal{N}(\boldsymbol{\mu}, \mathbf{I}_n)$ distribution, with expectation vector $\boldsymbol{\mu} = \mathbf{X}\boldsymbol{\beta}/\sigma$.

A direct application of the projection theorem shows that $(\boldsymbol{Y} - \boldsymbol{Y}^{(2)})/\sigma$ is independent of $\boldsymbol{Y}^{(2)}/\sigma$.

Since $\widehat{\boldsymbol{\beta}} = \mathbf{X}^{+}\mathbf{X}\widehat{\boldsymbol{\beta}} = \mathbf{X}^{+}\boldsymbol{Y}^{(2)}$ and $\widehat{\sigma^2} = \|\boldsymbol{Y} - \boldsymbol{Y}^{(2)}\|^2/n$, it follows that $\widehat{\sigma^2}$ is independent of $\widehat{\boldsymbol{\beta}}$.

Finally, by the same theorem, the random variable $\|\boldsymbol{Y} - \boldsymbol{Y}^{(2)}\|^2/\sigma^2$ has a χ^2_{n-p} distribution, as $\boldsymbol{Y}^{(2)}$ has the same expectation vector as $\boldsymbol{Y}$. □

As a corollary, we see that each estimator $\widehat{\beta}_i$ of β_i has a normal distribution with expectation β_i and variance $\sigma^2 \boldsymbol{u}_i^{\top}\mathbf{X}^{+}(\mathbf{X}^{+})^{\top}\boldsymbol{u}_i = \sigma^2\|\boldsymbol{u}_i^{\top}\mathbf{X}^{+}\|^2$, where $\boldsymbol{u}_i = [0, \ldots, 0, 1, 0, \ldots, 0]^{\top}$ is the i-th unit vector; in other words, the variance is $\sigma^2[(\mathbf{X}^{\top}\mathbf{X})^{+}]_{ii}$.

Hypothesis Testing

It is of interest to test the hypothesis $H_0 : \beta_i = 0$ against $H_1 : \beta_i \neq 0$, using the test statistic

$$T = \frac{\widehat{\beta}_i / \|\boldsymbol{u}_i^\top \mathbf{X}^+\|}{\sqrt{\text{RSE}}}, \tag{1}$$

where RSE is the residual squared error; that is $\text{RSE} = \text{RSS}/(n-p)$.

This test statistic has a t_{n-p} distribution under H_0.

To see this, write $T = Z/\sqrt{V/(n-p)}$, with

$$Z = \frac{\widehat{\beta}_i}{\sigma\|\boldsymbol{u}_i^\top \mathbf{X}^+\|} \quad \text{and} \quad V = n\,\widehat{\sigma^2}/\sigma^2.$$

By the previous theorem, $Z \sim \mathcal{N}(0,1)$ under H_0, $V \sim \chi^2_{n-p}$, and Z and V are independent. The result follows from the properties of the t distribution.

Comparing Two Normal Linear Models

Suppose $\boldsymbol{Y} = [Y_1, \ldots, Y_n]^\top$ is of the form

$$\boldsymbol{Y} = \underbrace{\mathbf{X}_1\boldsymbol{\beta}_1 + \mathbf{X}_2\boldsymbol{\beta}_2}_{\mathbf{X}\boldsymbol{\beta}} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}_n), \tag{2}$$

where $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are unknown vectors of dimension k and $p-k$, respectively; and $\mathbf{X}_1$ and $\mathbf{X}_2$ are full-rank model matrices of dimensions $n \times k$ and $n \times (p-k)$, respectively.

We wish to test the hypothesis $H_0 : \boldsymbol{\beta}_2 = \mathbf{0}$ against $H_1 : \boldsymbol{\beta}_2 \neq \mathbf{0}$.

Idea: compare the residual sum of squares for both models, expressed as $\|\boldsymbol{Y} - \boldsymbol{Y}^{(2)}\|^2$ and $\|\boldsymbol{Y} - \boldsymbol{Y}^{(1)}\|^2$.

Using Pythagoras' theorem we saw that
$\|\boldsymbol{Y} - \boldsymbol{Y}^{(2)}\|^2 - \|\boldsymbol{Y} - \boldsymbol{Y}^{(1)}\|^2 = \|\boldsymbol{Y}^{(2)} - \boldsymbol{Y}^{(1)}\|^2$

So we can base the decision whether to retain or reject H_0 on the basis of the quotient of $\|\boldsymbol{Y}^{(2)} - \boldsymbol{Y}^{(1)}\|^2$ and $\|\boldsymbol{Y} - \boldsymbol{Y}^{(2)}\|^2$.

Theorem: Test Statistic for Comp. Two Normal Linear Models

For the model (2), let $\boldsymbol{Y}^{(2)}$ and $\boldsymbol{Y}^{(1)}$ be the projections of $\boldsymbol{Y}$ onto the space spanned by the p columns of $\mathbf{X}$ and the k columns of $\mathbf{X}_1$, respectively. Then under $H_0 : \boldsymbol{\beta}_2 = \mathbf{0}$ the test statistic

$$T = \frac{\|\boldsymbol{Y}^{(2)} - \boldsymbol{Y}^{(1)}\|^2/(p-k)}{\|\boldsymbol{Y} - \boldsymbol{Y}^{(2)}\|^2/(n-p)} \tag{3}$$

has an $\mathsf{F}(p-k, n-p)$ distribution.

Proof

Define $\boldsymbol{X} := \boldsymbol{Y}/\sigma$ with expectation $\boldsymbol{\mu} := \mathbf{X}\boldsymbol{\beta}/\sigma$, and $\boldsymbol{X}_j := \boldsymbol{Y}^{(j)}/\sigma$ with expectation $\boldsymbol{\mu}_j$, $j = k, p$.

Note that $\boldsymbol{\mu}_p = \boldsymbol{\mu}$ and, under H_0, $\boldsymbol{\mu}_k = \boldsymbol{\mu}_p$. We can directly apply the projection theorem to find that

- $\|\boldsymbol{Y} - \boldsymbol{Y}^{(2)}\|^2/\sigma^2 = \|\boldsymbol{X} - \boldsymbol{X}_p\|^2 \sim \chi^2_{n-p}$ and,
- under H_0, $\|\boldsymbol{Y}^{(2)} - \boldsymbol{Y}^{(1)}\|^2/\sigma^2 = \|\boldsymbol{X}_p - \boldsymbol{X}_k\|^2 \sim \chi^2_{p-k}$.

Moreover, these random variables are independent of each other.

Finally, apply the following result from probability:

Theorem: Relationship Between χ^2 and F Distributions

Let $U \sim \chi^2_m$ and $V \sim \chi^2_n$ be independent. Then,

$$\frac{U/m}{V/n} \sim \mathsf{F}(m, n).$$

Hypothesis Testing for Nested Models

For nested models $[\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_k]$, $k = 1, 2, \ldots, d$, the F test can now be used to test whether certain $\mathbf{X}_i$ are needed or not. In particular, software packages will report the outcomes of

$$F_i = \frac{\|\boldsymbol{Y}^{(i)} - \boldsymbol{Y}^{(i-1)}\|/p_i}{\|\boldsymbol{Y} - \boldsymbol{Y}^{(d)}\|/(n-p)},$$

in the order $i = 2, 3, \ldots, d$.

Under the null hypothesis that $\boldsymbol{Y}^{(i)}$ and $\boldsymbol{Y}^{(i-1)}$ have the same expectation (that is, adding $\mathbf{X}_i$ to $\mathbf{X}_{i-1}$ has no additional effect on reducing the approximation error), the test statistic F_i has an $\mathsf{F}(p_i, n-p)$ distribution, and the corresponding P-values quantify the strength of the decision to include an additional variable in the model or not. This procedure is called analysis of variance (ANOVA).

Crop Yield (cont.)

We continue the crop yield example. Decompose the linear model as

$$\boldsymbol{Y} = \underbrace{\begin{bmatrix}\mathbf{1}\\ \mathbf{1}\\ \mathbf{1}\\ \mathbf{1}\end{bmatrix}}_{\mathbf{X}_1} \underbrace{\beta_0}_{\boldsymbol{\beta}_1} + \underbrace{\begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{1} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{1} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{1}\end{bmatrix}}_{\mathbf{X}_2} \underbrace{\begin{bmatrix}\beta_{12}\\ \beta_{13}\\ \beta_{14}\end{bmatrix}}_{\boldsymbol{\beta}_2} + \underbrace{\begin{bmatrix}\mathbf{C}\\ \mathbf{C}\\ \mathbf{C}\\ \mathbf{C}\end{bmatrix}}_{\mathbf{X}_3} \underbrace{\begin{bmatrix}\beta_{22}\\ \beta_{23}\\ \beta_{24}\end{bmatrix}}_{\boldsymbol{\beta}_3} + \boldsymbol{\varepsilon}.$$

Is the crop yield dependent on treatment levels as well as blocks?

We first test $\boldsymbol{\beta}_3 = \mathbf{0}$ versus $\boldsymbol{\beta}_3 \neq \mathbf{0}$.

$\boldsymbol{Y}^{(2)}$ is the projection of $\boldsymbol{Y}$ onto the $(p = 7)$-dimensional space spanned by the columns of $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3]$.

$\boldsymbol{Y}^{(1)}$ is the projection of $\boldsymbol{Y}$ onto the $(k = 4)$-dimensional space spanned by the columns of $\mathbf{X}_{12} := [\mathbf{X}_1, \mathbf{X}_2]$

The test statistic, T_{12} say, under H_0 has an $\mathsf{F}(3, 9)$ distribution.

Crop Yield (cont.)

The Python code below calculates the outcome of the test statistic T_{12} and the corresponding P-value. We find $t_{12} = 34.9998$, which gives a P-value 2.73×10^{-5}. This shows that the block effects are extremely important for explaining the data.

Using the extended model (including the block effects), we can test whether $\boldsymbol{\beta}_2 = \mathbf{0}$ or not; that is, whether the treatments have a significant effect on the crop yield in the presence of the Block factor.

This is done in the last six lines of the code below. The outcome of the test statistic is 4.4878, with a P-value of 0.0346.

By including the block effects, we effectively reduce the uncertainty in the model and are able to more accurately assess the effects of the treatments, to conclude that the treatment seems to have an effect on the crop yield.

crop.py

```
import numpy as np
from scipy.stats import f
from numpy.linalg import lstsq, norm

yy = np.array([9.2988, 9.4978, 9.7604, 10.1025,
      8.2111, 8.3387, 8.5018,  8.1942,
      9.0688, 9.1284, 9.3484,  9.5086,
      8.2552, 7.8999, 8.4859,  8.9485]).reshape(4,4).T

nrow, ncol = yy.shape[0], yy.shape[1]
n = nrow * ncol
y = yy.reshape(16,)
X_1 = np.ones((n,1))

KM = np.kron(np.eye(ncol),np.ones((nrow,1)))
KM[:,0]
X_2 = KM[:,1:ncol]
IM = np.eye(nrow)
C = IM[:,1:nrow]

X_3 = np.vstack((C, C))
X_3 = np.vstack((X_3, C))
X_3 = np.vstack((X_3, C))
```

```
X = np.hstack((X_1,X_2))
X = np.hstack((X,X_3))

p = X.shape[1] #number of parameters in full model
betahat = lstsq(X, y,rcond=None)[0]  #estimate under the full model

ym = X @ betahat

X_12 = np.hstack((X_1, X_2)) #omitting the block effect
k = X_12.shape[1] #number of parameters in reduced model
betahat_12 = lstsq(X_12, y,rcond=None)[0]
y_12 = X_12 @ betahat_12
T_12=(n-p)/(p-k)*(norm(y-y_12)**2 -  norm(y-ym)**2)/norm(y-ym)**2
pval_12 = 1 - f.cdf(T_12,p-k,n-p)

X_13 = np.hstack((X_1, X_3)) #omitting the treatment effect
k = X_13.shape[1] #number of parameters in reduced model
betahat_13 = lstsq(X_13, y,rcond=None)[0]
y_13 = X_13 @ betahat_13
T_13=(n-p)/(p-k)*(norm(y-y_13)**2 - norm(y-ym)**2)/norm(y-ym)**2
pval_13 = 1 - f.cdf(T_13,p-k,n-p)
```

Confidence and Prediction Intervals

Suppose we wish to predict how a new response variable will behave on the basis of a new explanatory vector $\boldsymbol{x}$.

Thus, consider a new $\boldsymbol{x}$ and let $Y \sim \mathcal{N}(\boldsymbol{x}^\top \boldsymbol{\beta}, \sigma^2)$, with $\boldsymbol{\beta}$ and σ^2 unknown.

First we are going to look at the *expected* value $\mathbb{E}Y = \boldsymbol{x}^\top \boldsymbol{\beta}$.

This can estimate it via $\widehat{Y} = \boldsymbol{x}^\top \widehat{\boldsymbol{\beta}}$, where $\widehat{\boldsymbol{\beta}} \sim \mathcal{N}(\boldsymbol{\beta},\ \sigma^2(\mathbf{X}^\top \mathbf{X})^+)$.

It follows that $\widehat{Y} \sim \mathcal{N}(\boldsymbol{x}^\top \boldsymbol{\beta},\ \sigma^2 \|\boldsymbol{x}^\top \mathbf{X}^+\|^2)$.

Let $Z \sim \mathcal{N}(0, 1)$ be the standardized version of $\widehat{Y}$ and $V = \|\boldsymbol{Y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|^2/\sigma^2 \sim \chi^2_{n-p}$. Then the random variable

$$T := \frac{(\boldsymbol{x}^\top \widehat{\boldsymbol{\beta}} - \boldsymbol{x}^\top \boldsymbol{\beta}) \,/\, \|\boldsymbol{x}^\top \mathbf{X}^+\|}{\|\boldsymbol{Y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\| \,/\sqrt{(n-p)}} = \frac{Z}{\sqrt{V/(n-p)}}$$

has a t_{n-p} distribution.

Confidence Interval for the Expected Response

After rearranging the identity $\mathbb{P}(|T| \leqslant t_{n-p;1-\alpha/2}) = 1 - \alpha$, where $t_{n-p;1-\alpha/2}$ is the $(1 - \alpha/2)$ quantile of the t_{n-p} distribution, we arrive at the stochastic confidence interval

$$\boldsymbol{x}^\top \widehat{\boldsymbol{\beta}} \pm t_{n-p;1-\alpha/2} \sqrt{\mathrm{RSE}}\, \|\boldsymbol{x}^\top \mathbf{X}^+\|,$$

where we have identified $\|\boldsymbol{Y} - \mathbf{X}\widehat{\boldsymbol{\beta}}\|^2/(n-p)$ with RSE.

This confidence interval quantifies the uncertainty in the learner (regression surface).

Prediction Interval

Construct an interval such that Y lies in this interval with a certain guaranteed probability. We have *two* sources of variation:

1. $Y \sim \mathcal{N}(\boldsymbol{x}^\top \boldsymbol{\beta}, \sigma^2)$ itself is a random variable.
2. Estimating $\boldsymbol{x}^\top \boldsymbol{\beta}$ via $\widehat{Y}$ brings another source of variation.

Since $Y \sim \mathcal{N}(\boldsymbol{x}^\top \boldsymbol{\beta}, \sigma^2)$ and $\widehat{Y} \sim \mathcal{N}(\boldsymbol{x}^\top \boldsymbol{\beta}, \sigma^2 \|\boldsymbol{x}^\top \mathbf{X}^+\|^2)$ are independent, it follows that $Y - \widehat{Y} \sim \mathcal{N}(0, \sigma^2(1 + \|\boldsymbol{x}^\top \mathbf{X}^+\|^2)$.

Letting $Z \sim \mathcal{N}(0, 1)$ be the standardized version of $Y - \widehat{Y}$, and repeating the steps used for the construction of the confidence interval, we arrive at the prediction interval

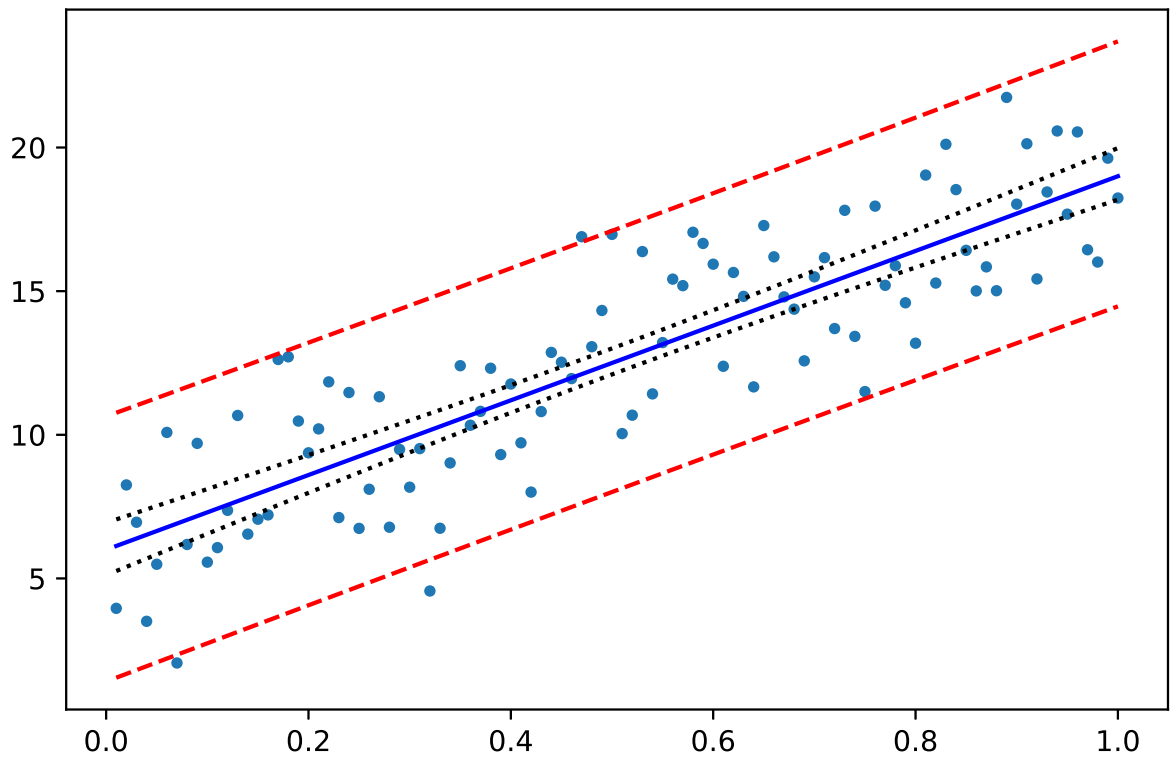
$$\boldsymbol{x}^\top \widehat{\boldsymbol{\beta}} \pm t_{n-p;1-\alpha/2} \sqrt{\text{RSE}} \sqrt{1 + \|\boldsymbol{x}^\top \mathbf{X}^+\|^2}.$$

This prediction interval captures the uncertainty from an as-yet-unobserved response as well as the uncertainty in the parameters of the regression model itself.

Example: Confidence Limits in Linear Regression

The data ($n = 100$ samples) are drawn from a simple linear regression model with parameters $\boldsymbol{\beta} = [6, 13]^\top$ and $\sigma = 2$, where the x-coordinates are evenly spaced on the interval $[0, 1]$.

The figure shows 95% numeric confidence and prediction curves as well as the true regression line.



confpred.py

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import t
from  numpy.linalg import inv, lstsq, norm
np.random.seed(123)

n = 100
x = np.linspace(0.01,1,100).reshape(n,1)
# parameters
beta = np.array([6,13])
sigma = 2
Xmat = np.hstack((np.ones((n,1)), x)) #design matrix
y = Xmat @ beta + sigma*np.random.randn(n)

# solve the normal equations
betahat = lstsq(Xmat, y,rcond=None)[0]
# estimate for sigma
sqMSE = norm(y - Xmat @ betahat)/np.sqrt(n-2)
tquant = t.ppf(0.975,n-2) # 0.975 quantile
ucl = np.zeros(n) #upper conf. limits
lcl = np.zeros(n) #lower conf. limits
upl = np.zeros(n)
lpl = np.zeros(n)
```

```
rl = np.zeros(n)  # (true) regression line
u = 0

for i in range(n):
    u = u + 1/n;
    xvec = np.array([1,u])
    sqc = np.sqrt(xvec.T @ inv(Xmat.T @ Xmat) @ xvec)
    sqp = np.sqrt(1 + xvec.T @ inv(Xmat.T @ Xmat) @ xvec)
    rl[i] = xvec.T @ beta;
    ucl[i] = xvec.T @ betahat + tquant*sqMSE*sqc;
    lcl[i] = xvec.T @ betahat - tquant*sqMSE*sqc;
    upl[i] = xvec.T @ betahat + tquant*sqMSE*sqp;
    lpl[i] = xvec.T @ betahat - tquant*sqMSE*sqp;

plt.plot(x,y, '.')
plt.plot(x,rl,'b')
plt.plot(x,ucl,'k:')
plt.plot(x,lcl,'k:')
plt.plot(x,upl,'r--')
plt.plot(x,lpl,'r--')
```