Linear regression

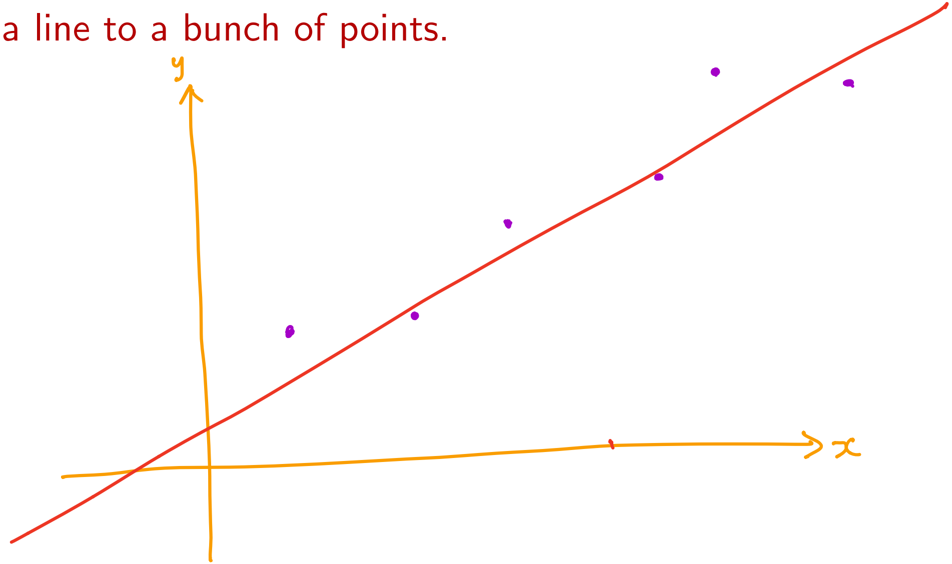
DSE 220

Overview

1. Introduction to linear regression
2. Multivariate least-squares regression
3. Regularized regression

Linear regression

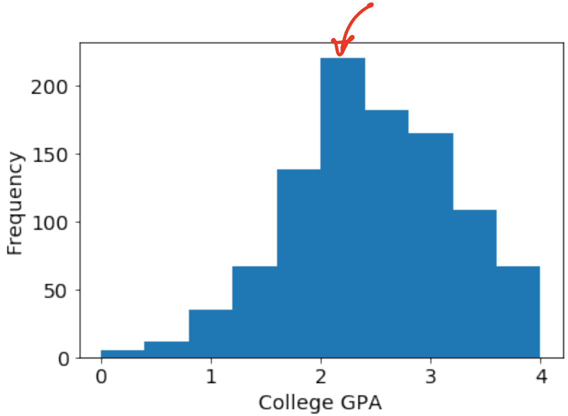
Fitting a line to a bunch of points.



How do we do this?

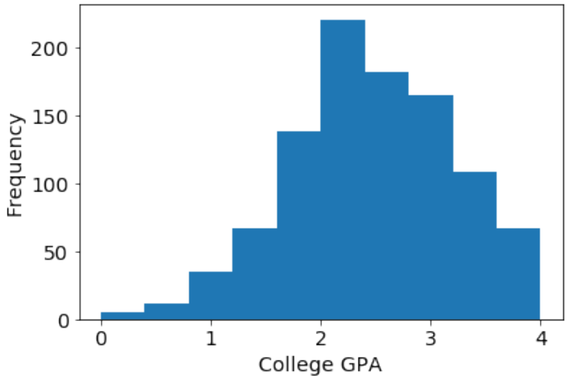
Example: college GPAs

Distribution of GPAs of students at a certain Ivy League university.



Example: college GPAs

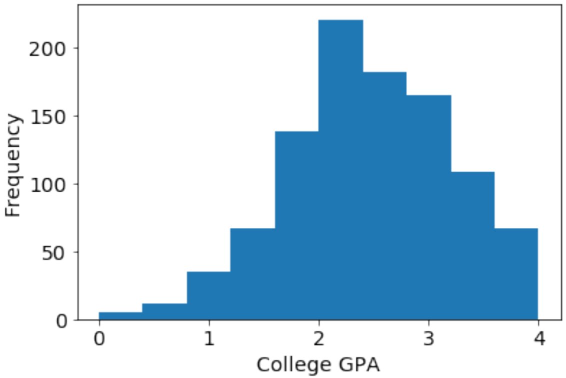
Distribution of GPAs of students at a certain Ivy League university.



What GPA to predict for a random student from this group?

Example: college GPAs

Distribution of GPAs of students at a certain Ivy League university.



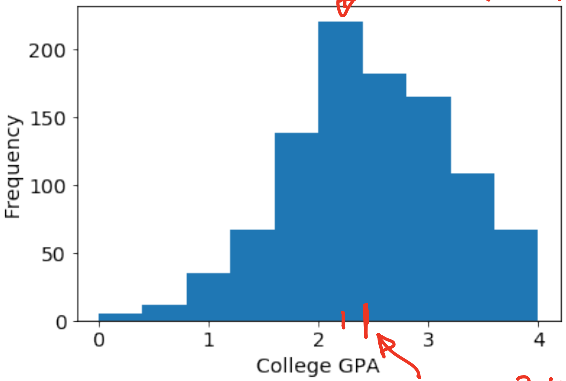
What GPA to predict for a random student from this group?

- Without further information, predict the **mean**, 2.47.

Example: college GPAs

Distribution of GPAs of students at a certain Ivy League university.

most common value 2.25 (mode)



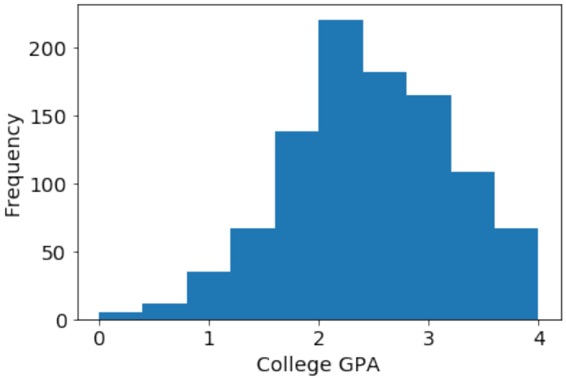
mean = 2.47

What GPA to predict for a random student from this group?

- Without further information, predict the **mean**, 2.47. ← In what sense is this a good choice?
- What is the average squared error of this prediction? That is, $\underbrace{\mathbb{E}[((\text{student's GPA}) - (\text{predicted GPA}))^2]}_{\text{squared error in our prediction}}$? ← average (mean) squared error (MSE)

Example: college GPAs

Distribution of GPAs of students at a certain Ivy League university.

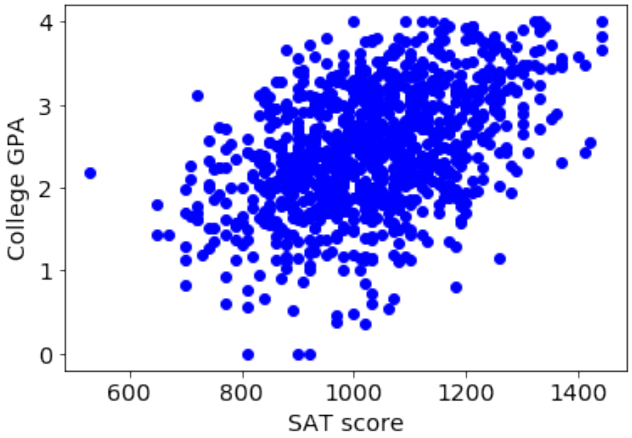


What GPA to predict for a random student from this group?

- Without further information, predict the **mean**, 2.47.
- What is the average squared error of this prediction?
  That is, $\mathbb{E}[((\text{student's GPA}) - (\text{predicted GPA}))^2]$?
  The **variance** of the distribution, 0.55.

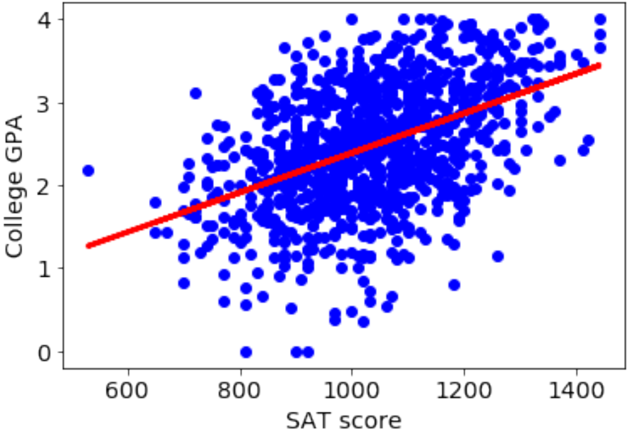
Better predictions with more information

We also have SAT scores of all students.



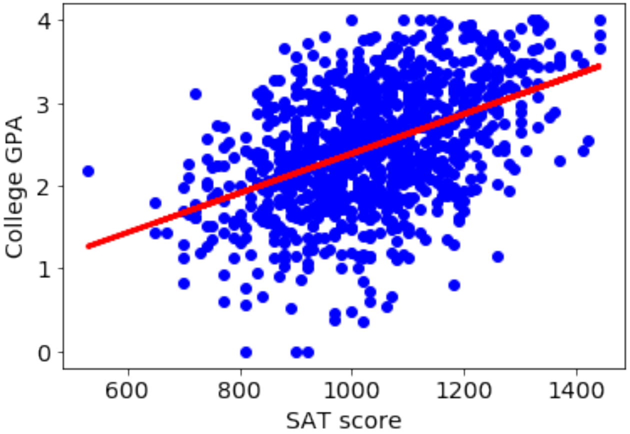
Better predictions with more information

We also have SAT scores of all students.



Better predictions with more information

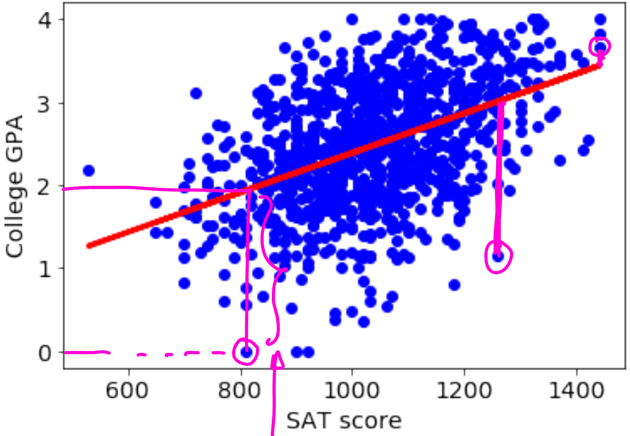
We also have SAT scores of all students.



Mean squared error (MSE) drops to 0.43.

Better predictions with more information

We also have SAT scores of all students.



squared error for this one person

Mean squared error (MSE) drops to 0.43.

average of all these individual squared errors

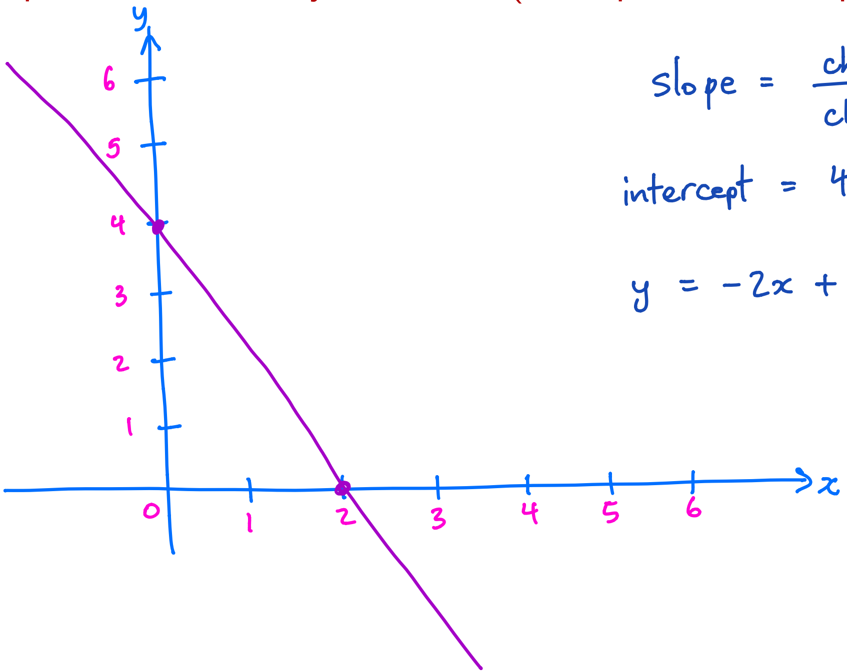
This is a **regression** problem with:

- **Predictor variable**: SAT score ← information used for prediction
- **Response variable**: College GPA ← the thing we're predicting

Parametrizing a line

In $\mathbb{R}^2$, a line has <u>two</u> parameters

A line can be parameterized as $y = ax + b$ (a: slope, b: intercept).



$$\text{slope} = \frac{\text{change in } y}{\text{change in } x} = \frac{-4}{2} = -2$$

$$\text{intercept} = 4$$

$$y = -2x + 4$$

The line fitting problem

Root MSE $= \sqrt{MSE}$

Pick a line (a, b) based on $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)}) \in \mathbb{R} \times \mathbb{R}$

- $x^{(i)}, y^{(i)}$ are predictor and response variables.
  E.g. SAT score, GPA of ith student.
- Minimize the mean squared error,

actual response for $x^{(i)}$

prediction made by line on $x^{(i)}$

$$\mathrm{MSE}(a, b) = \frac{1}{n}\sum_{i=1}^{n} \boxed{(y^{(i)} - (ax^{(i)} + b))^2}.$$

squared error of prediction on $x^{(i)}$

This is the **loss function**.

We are formulating a LEARNING problem (ie. learn a good linear predictor) as an OPTIMIZATION task: find the parameters (a, b) that MINIMIZE this LOSS FUNCTION.

Minimizing the loss function

$$\frac{d}{da}(u^2) = 2u\,\frac{du}{da}$$

Given $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$, minimize

call this u

$$L(a,b) = \sum_{i=1}^{n} \underbrace{(y^{(i)} - (ax^{(i)} + b))^2}_{\text{this is } u^2}.$$

To minimize, set $\frac{dL}{da} = \frac{dL}{db} = 0.$

$$\frac{dL}{da} = \sum_{i=1}^{n} 2\left(y^{(i)} - (ax^{(i)} + b)\right)\left(-x^{(i)}\right) = -2\sum_{i=1}^{n}\left(y^{(i)} - (ax^{(i)}+b)\right)x^{(i)}$$

$$\frac{dL}{db} = \sum_{i=1}^{n} 2\left(y^{(i)} - (ax^{(i)} + b)\right)(-1) = -2\sum_{i=1}^{n}\left(y^{(i)} - (ax^{(i)}+b)\right)$$

$$\frac{dL}{db} = 0 \implies \sum_{i=1}^{n} \left(y^{(i)} - ax^{(i)} - b\right) = 0$$

$$\left(\sum_{i=1}^{n} \left(y^{(i)} - ax^{(i)}\right)\right) - nb = 0$$

$$\implies b = \frac{1}{n} \sum_{i=1}^{n} \left(y^{(i)} - ax^{(i)}\right) = \frac{1}{n} \sum_{i=1}^{n} y^{(i)} - a \cdot \frac{1}{n} \sum_{i=1}^{n} x^{(i)}$$

this is the avg. y-value; call it $\bar{y}$

avg. x-value; call it $\bar{x}$

$$\implies b = \bar{y} - a\bar{x}$$

$$\frac{dL}{da} = 0 \implies a = \frac{\sum_{i=1}^{n} \left(y^{(i)} - \bar{y}\right)\left(x^{(i)} - \bar{x}\right)}{\sum_{i=1}^{n} \left(x^{(i)} - \bar{x}\right)^2}$$

a kind of "average slope"

Closed-form solutions for <u>a</u> and <u>b</u> !

Overview

1. Introduction to linear regression
2. Multivariate least-squares regression
3. Regularized regression

Multivariate regression: diabetes study

Data from $n = 442$ diabetes patients.

For each patient:

- 10 features $x = (x_1, \ldots, x_{10})$

  age, sex, body mass index, average blood pressure, and six blood serum measurements.

- A real value y: the progression of the disease a year later.

Regression problem:

- **response** $y \in \mathbb{R}$
- **predictor variables** $x \in \mathbb{R}^{10}$

Least-squares regression

Linear function of 10 variables: for $x \in \mathbb{R}^{10}$,

$$f(x) = w_1 x_1 + w_2 x_2 + \cdots + w_{10} x_{10} + b = w \cdot x + b$$

where $w = (w_1, w_2, \ldots, w_{10})$.

Least-squares regression

Linear function of 10 variables: for $x \in \mathbb{R}^{10}$,

$$f(x) = w_1 x_1 + w_2 x_2 + \cdots + w_{10} x_{10} + b = w \cdot x + b$$

where $w = (w_1, w_2, \ldots, w_{10})$.

actual value

Penalize error using **squared loss** $(y - \underbrace{(w \cdot x + b)}_{\text{prediction on } x})^2$.

Least-squares regression

Linear function of 10 variables: for $x \in \mathbb{R}^{10}$,

$$f(x) = w_1 x_1 + w_2 x_2 + \cdots + w_{10} x_{10} + b = w \cdot x + b$$

where $w = (w_1, w_2, \ldots, w_{10})$.

Penalize error using **squared loss** $(y - (w \cdot x + b))^2$.

$n = 442$

$d = 10$

Least-squares regression:

- *Given:* data $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)}) \in \mathbb{R}^d \times \mathbb{R}$
- *Return:* linear function given by $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$ ← parameters to be learned (11 params)
- *Goal:* minimize the **loss function**

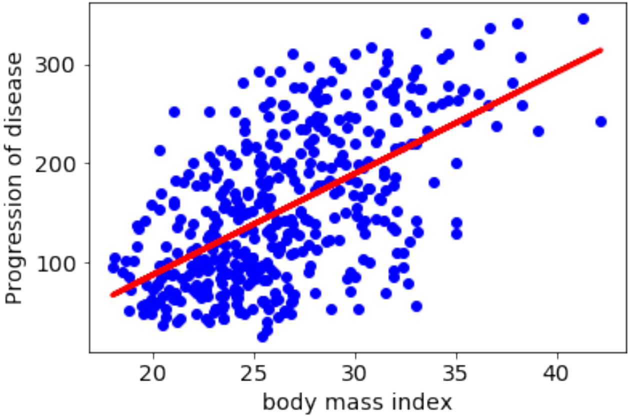
$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2.$$ total squared error

Back to the diabetes data

- No predictor variables: mean squared error (MSE) $= 5930$

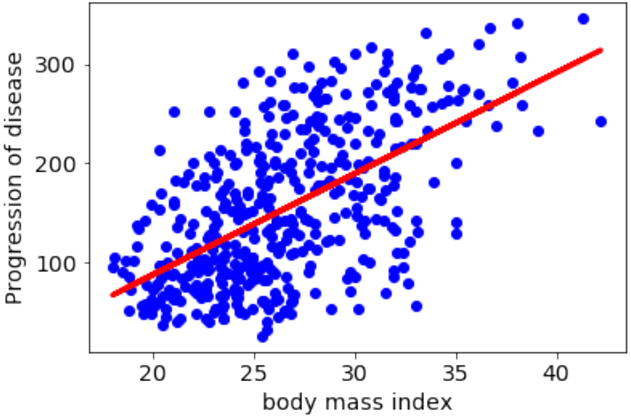
Back to the diabetes data

- No predictor variables: mean squared error (MSE) $= 5930$
- One predictor ('bmi'): MSE $= 3890$



Back to the diabetes data

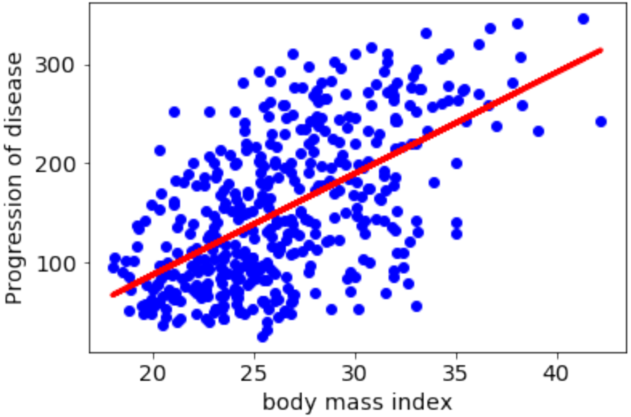
- No predictor variables: mean squared error (MSE) = 5930
- One predictor ('bmi'): MSE = 3890



- Two predictors ('bmi', 'serum5'): MSE = 3205

Back to the diabetes data

- No predictor variables: mean squared error (MSE) $= 5930$
- One predictor ('bmi'): MSE $= 3890$



- Two predictors ('bmi', 'serum5'): MSE $= 3205$
- All ten predictors: MSE $= 2860$

Least-squares solution 1

$(w_1, \ldots, w_d, b)$

Linear function of d variables given by $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$:

$(b, w_1, \ldots, w_d)$

$$f(x) = w_1 x_1 + w_2 x_2 + \cdots + w_d x_d + b = w \cdot x + b$$

$(1, (x_1, \ldots x_d)$

$, 1)$

Least-squares solution 1

Linear function of d variables given by $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$:

annoying

$$f(x) = w_1 x_1 + w_2 x_2 + \cdots + w_d x_d + b = w \cdot x + b$$

Assimilate the intercept b into w:

- Add a new feature that is identically 1: let $\widetilde{x} = (1, x) \in \mathbb{R}^{d+1}$

$$\underbrace{\overset{x}{\begin{pmatrix} 4 & 0 & 2 & \cdots & 3 \end{pmatrix}}}_{d} \Longrightarrow \underbrace{\overset{\widetilde{x}}{\begin{pmatrix} 1 & 4 & 0 & 2 & \cdots & 3 \end{pmatrix}}}_{d+1}$$

- Set $\widetilde{w} = (b, w) \in \mathbb{R}^{d+1}$
- Then $f(x) = w \cdot x + b = \widetilde{w} \cdot \widetilde{x}$

$w \cdot x + b$

$w_1 x_1 + \cdots + w_d x_d + b = \underbrace{\begin{pmatrix} 1 \\ x_1 \\ \vdots \\ x_d \end{pmatrix}}_{\widetilde{x}} \cdot \underbrace{\begin{pmatrix} b \\ w_1 \\ \vdots \\ w_d \end{pmatrix}}_{\widetilde{w}}$ $\updownarrow d+1$

Goal: find $\widetilde{w} \in \mathbb{R}^{d+1}$ that minimizes

data

$$L(\widetilde{w}) = \sum_{i=1}^{n} \left(y^{(i)} - \widetilde{w} \cdot \widetilde{x}^{(i)} \right)^2$$

parameter to "learn"

Least-squares solution 2

$$\underset{(d+1)\times 1}{X\tilde{w}} = \underset{n\times(d+1)}{\begin{pmatrix} \text{---} & \tilde{x}^{(1)} & \text{---} \\ & \vdots & \\ \text{---} & \tilde{x}^{(n)} & \text{---} \end{pmatrix}} \underset{(d+1)\times 1}{\begin{pmatrix} | \\ \tilde{w} \\ | \end{pmatrix}} = \begin{pmatrix} \tilde{w}\cdot\tilde{x}^{(1)} \\ \tilde{w}\cdot\tilde{x}^{(2)} \\ \vdots \\ \tilde{w}\cdot\tilde{x}^{(n)} \end{pmatrix} \updownarrow n$$

vector of n predictions

Write

$$\underset{n\times(d+1)}{X} = \begin{pmatrix} \longleftarrow & \tilde{x}^{(1)} & \longrightarrow \\ \longleftarrow & \tilde{x}^{(2)} & \longrightarrow \\ & \vdots & \\ \longleftarrow & \tilde{x}^{(n)} & \longrightarrow \end{pmatrix}, \quad \underset{n\times 1}{y} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ \vdots \\ y^{(n)} \end{pmatrix}$$

$\longleftarrow d+1 \longrightarrow$

$$\underbrace{\begin{pmatrix} y^{(1)} - \tilde{w}\cdot\tilde{x}^{(1)} \\ y^{(2)} - \tilde{w}\cdot\tilde{x}^{(2)} \\ \vdots \\ y^{(n)} - \tilde{w}\cdot\tilde{x}^{(n)} \end{pmatrix}}_{\text{vector of } n \text{ errors}}$$

Then the loss function is

$$L(\tilde{w}) = \sum_{i=1}^{n} (y^{(i)} - \tilde{w}\cdot\tilde{x}^{(i)})^2 = \|y - X\tilde{w}\|^2$$

if this is not invertible, we use the "pseudoinverse"

and it is minimized at $\tilde{w} = (\underbrace{X^T X}_{(d+1)\times(d+1)})^{-1}(X^T y)$. $\leftarrow n$ } Can get this by calculus.

Overview

1. Introduction to linear regression
2. Multivariate least-squares regression
3. Regularized regression

Generalization behavior of least-squares regression

Given a **training set** $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)}) \in \mathbb{R}^d \times \mathbb{R}$, find a linear function, given by $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$, that minimizes the squared loss

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2.$$

Is training loss a good estimate of **future** performance?

Generalization behavior of least-squares regression

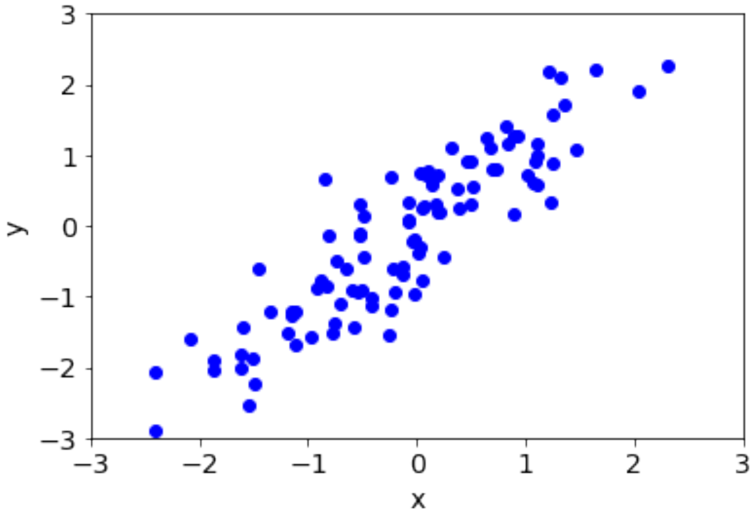
Given a **training set** $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)}) \in \mathbb{R}^d \times \mathbb{R}$, find a linear function, given by $w \in \mathbb{R}^d$ and $b \in \mathbb{R}$, that minimizes the squared loss

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2.$$

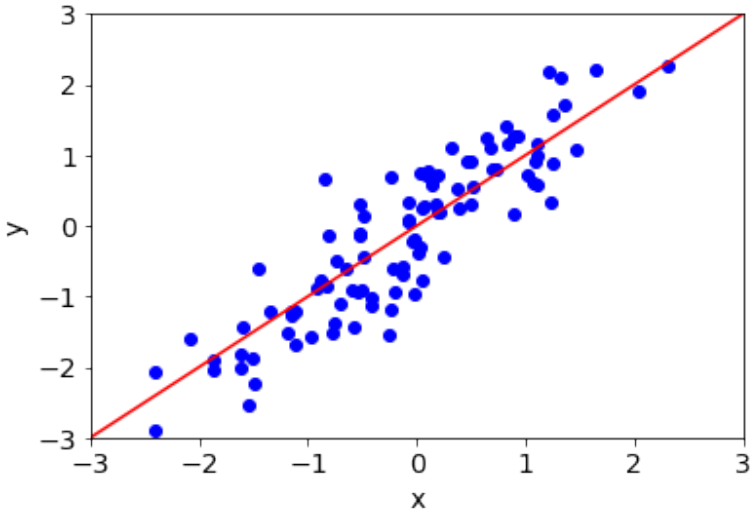
Is training loss a good estimate of **future** performance?

- If n is large enough: maybe.
- Otherwise: probably an underestimate.

Example

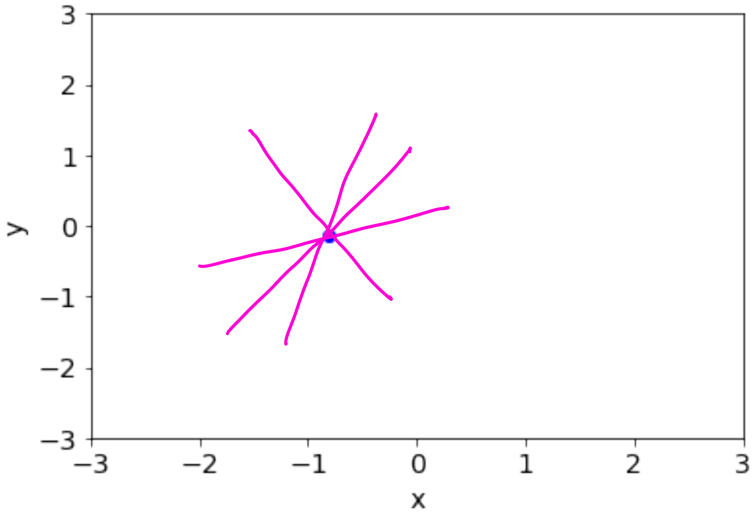


Example

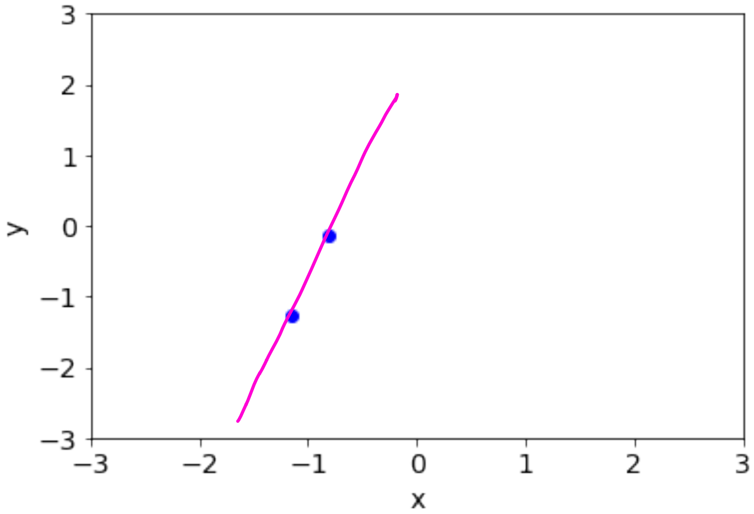


Lots of data (considering $d=1$) and training error is probably a pretty indication of test error.

Example



Example



Better error estimates

Recall: k-**fold cross-validation**

- Divide the data set into k equal-sized groups $S_1, \ldots, S_k$
- For $i = 1$ to k:
    - Train a regressor on all data except S_i
    - Let E_i be its error on S_i
- Error estimate: average of $E_1, \ldots, E_k$

Better error estimates

Recall: k-**fold cross-validation**

- Divide the data set into k equal-sized groups $S_1, \ldots, S_k$
- For $i = 1$ to k:
  - Train a regressor on all data except S_i
  - Let E_i be its error on S_i
- Error estimate: average of $E_1, \ldots, E_k$

A nagging question:
When n is small, should we be minimizing the squared loss?

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2$$

Ridge regression

Minimize squared loss **plus** a term that penalizes “complex” w:

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|^2$$

Adding a penalty term like this is called **regularization**.

Ridge regression

Minimize squared loss **plus** a term that penalizes "complex" w:

regularizer

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|^2$$

what is effect of λ?

Adding a penalty term like this is called **regularization**.

Put predictor vectors in matrix X and responses in vector y:

$$w = (X^T X + \lambda I)^{-1} (X^T y)$$

Least-squares: $\lambda = 0$

$\underline{\lambda = 0}$

Regular least-squares

Reasonable when we have a lot of data

$\underline{\lambda \to \infty}$

Only the second term matters

Solution: $w \to 0$

Reasonable when we have <u>no</u> data

Pick an intermediate λ between these.

Essentially shrinks the least-squares solution towards zero.

SHRINKAGE ESTIMATOR

Toy example

Training, test sets of 100 points

- $x \in \mathbb{R}^{100}$, each feature x_i is Gaussian $N(0, 1)$
- $y = x_1 + \cdots + x_{10} + N(0, 1)$

Toy example

Training, test sets of 100 points

- $x \in \mathbb{R}^{100}$, each feature x_i is Gaussian $N(0,1)$
- $y = x_1 + \cdots + x_{10} + N(0,1)$

λ	training MSE	test MSE
0.00001	0.00	585.81
0.0001	0.00	564.28
0.001	0.00	404.08
0.01	0.01	83.48
0.1	0.03	19.26
1.0	0.07	7.02
10.0	0.35	2.84
100.0	2.40	5.79
1000.0	8.19	10.97
10000.0	10.83	12.63

$\lambda \approx 0$, least-squares estimate

sweet spot somewhere in here
find it using cross-validation

$w \approx 0$, test MSE $\approx$ variance of y

The lasso

Popular "shrinkage" estimators:

- **Ridge regression**

$$L(w,b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|_2^2$$

- **Lasso**: tends to produce sparse w

$$L(w,b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|_1$$

Why would we want a sparse solution w?

① Generalize better by eliminating irrelevant features

② Less space

③ Easier to understand

The lasso

Popular "shrinkage" estimators:

- **Ridge regression**

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|_2^2$$

- **Lasso**: tends to produce sparse w

$$L(w, b) = \sum_{i=1}^{n} (y^{(i)} - (w \cdot x^{(i)} + b))^2 + \lambda \|w\|_1$$

Toy example:
Lasso recovers 10 relevant features plus a few more.