MATH 829: Introduction to Data Mining and Analysis
Linear Regression: old and new

Dominique Guillot

Departments of Mathematical Sciences
University of Delaware

February 10, 2016

Linear Regression: old and new

- Typical problem: we are given n observations of variables $X_1, \dots, X_p$ and Y.
- **Goal**: Use $X_1, \dots, X_p$ to try to predict Y.
- Example: Cars data compiled using Kelley Blue Book ($n = 805, p = 11$).

Price	Mileage	Make	Model	Trim	Type	Cylinder	Liter	Doors	Cruise	Sound	Leather
17314.103	8221	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	1
17542.036	9135	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0
16218.848	13196	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0
16336.913	16342	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	0	0
16339.17	19832	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	0	1
15709.053	22236	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0
15230	22576	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0
15048.042	22964	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	0
14862.094	24021	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	0	1
15295.018	27325	Buick	Century	Sedan 4D	Sedan	6	3.1	4	1	1	1
21335.852	10237	Buick	Lacrosse	CX Sedan	Sedan	6	3.6	4	1	0	0
20538.088	15066	Buick	Lacrosse	CX Sedan	Sedan	6	3.6	4	1	1	0
20512.094	16633	Buick	Lacrosse	CX Sedan	Sedan	6	3.6	4	1	1	0
19924.159	19800	Buick	Lacrosse	CX Sedan	Sedan	6	3.6	4	1	1	1
19774.249	23359	Buick	Lacrosse	CX Sedan	Sedan	6	3.6	4	1	1	1
19344.166	23765	Buick	Lacrosse	CX Sedan	Sedan	6	3.6	4	1	1	0

- Find a **linear model** $Y = \beta_1 X_1 + \dots + \beta_p X_p$.
- In the example, we want:
  $\text{price} = \beta_1 \cdot \text{mileage} + \beta_2 \cdot \text{cylinder} + \dots$

Linear regression: classical setting

$p = \text{nb. of variables}, n = \text{nb. of observations}.$

Classical setting:

- $n \gg p$ (n much larger than p). With enough observations, we hope to be able to build a good model.
- Note: even if the "true" relationship between the variables is not linear, we can include **transformations** of variables.
- E.g.

$$X_{p+1} = X_1^2, X_{p+2} = X_2^2, \dots$$

- Note: adding transformed variables can increase p significantly.
- A complex model requires a lot of observations.

Modern setting:

- In modern problems, it is often the case that $n \ll p$.
- Requires supplementary assumptions (e.g. sparsity).
- Can still build good models with very few observations.

Classical setting

Idea:

$$Y \in \mathbb{R}^{n\times 1} \qquad X \in \mathbb{R}^{n\times p}$$

$$Y = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix} \qquad X = \begin{pmatrix} | & | & \dots & | \\ \mathbf{x_1} & \mathbf{x_2} & \dots & \mathbf{x_p} \\ | & | & \dots & | \end{pmatrix},$$

where $\mathbf{x_1}, \dots, \mathbf{x_p} \in \mathbb{R}^{n\times 1}$ are the observations of $X_1, \dots X_p$.

- We want $Y = \beta_1 X_1 + \dots + \beta_p X_p$.
- Equivalent to solving

$$Y = X\beta \qquad \beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{pmatrix}.$$

Classical setting (cont.)

We need to solve $Y = X\beta$.

- Obviously, in general, the system has **no solution**.
- A popular approach is to solve the system in the least squares sense:

$$\hat{\beta} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|Y - X\beta\|^2.$$

- How do we compute the solution?

Calculus approach:

$$\begin{aligned}\frac{\partial}{\partial \beta_i}\|Y - X\beta\|^2 &= \frac{\partial}{\partial \beta_i}\sum_{k=1}^{n}(y_i - X_{k1}\beta_1 - X_{k2}\beta_2 - \cdots - X_{kp}\beta_p)^2 \\ &= 2\sum_{k=1}^{n}(y_i - X_{k1}\beta_1 - X_{k2}\beta_2 - \cdots - X_{kp}\beta_p) \times (-X_{ki}) \\ &= 0.\end{aligned}$$

Therefore,

$$\sum_{k=1}^{n} X_{ki}(X_{k1}\beta_1 + X_{k2}\beta_2 + \cdots + X_{kp}\beta_p) = \sum_{k=1}^{n} X_{ki}y_i$$

Calculus approach (cont.)

Now

$$\sum_{k=1}^{n} X_{ki}(X_{k1}\beta_1 + X_{k2}\beta_2 + \cdots + X_{kp}\beta_p) = \sum_{k=1}^{n} X_{ki}y_i \qquad i = 1, \ldots, p,$$

is equivalent to:

$$X^TX\beta = X^Ty \qquad \text{(Normal equations)}.$$

We compute the Hessian:

$$\frac{\partial^2}{\partial \beta_i \beta_j}\|Y - X\beta\|^2 = 2X^TX.$$

If X^TX is invertible, then X^TX is positive definite and

$$\hat{\beta} = (X^TX)^{-1}X^TY$$

is the unique minimum of $\|Y - X\beta\|^2$.

Linear algebra approach

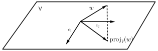
Want to solve $Y = X\beta$.

Linear algebra approach: Recall: If $V \subset \mathbb{R}^n$ is a subspace and $w \notin V$, then the best approximation of w be a vector in V is

$$\text{proj}_V(w).$$

"Best" in the sense that:

$$\|w - \text{proj}_V(w)\| \leq \|w - v\| \qquad \forall v \in V.$$



Here:

$$X\beta \in \text{col}(X) = \text{span}(\mathbf{x_1}, \ldots, \mathbf{x_p}).$$

If $Y \notin \text{col}(X)$, then the best approximation of Y by a vector in $\text{col}(X)$ is

$$\text{proj}_{\text{col}(X)}(Y).$$

Linear algebra approach (cont.)

So
$$\|Y - \text{proj}_{\text{col}(X)}(Y)\| \leq \|Y - X\beta\| \qquad \forall \beta \in \mathbb{R}^p.$$

Therefore, to find $\hat{\beta}$, we solve

$$X\hat{\beta} = \text{proj}_{\text{col}(X)}(Y)$$

(Note: this system always has a solution.)

With a little more work, we can find an explicit solution:

$$Y - X\hat{\beta} = Y - \text{proj}_{\text{col}(X)}(Y) = \text{proj}_{\text{col}(X)^\perp}(Y).$$

Recall

$$\text{col}(X)^\perp = \text{null}(X^T).$$

Thus,

$$Y - X\hat{\beta} = \text{proj}_{\text{null}(X^T)}(Y) \in \text{null}(X^T).$$

That implies:

$$X^T(Y - X\hat{\beta}) = 0.$$

Equivalently,

$$X^TX\hat{\beta} = X^TY \qquad \text{(Normal equations)}.$$

The least squares theorem

Theorem (Least squares theorem)

Let $A \in \mathbb{R}^{n \times m}$ and $b \in \mathbb{R}^n$. Then

1. *$Ax = b$ always has a least squares solution $\hat{x}$.*
2. *A vector $\hat{x}$ is a least squares solution iff it satisfies the normal equations*
$$A^T A\hat{x} = A^T b.$$
3. *$\hat{x}$ is unique $\Leftrightarrow$ the columns of A are linearly independent $\Leftrightarrow$ $A^T A$ is invertible. In that case, the unique least squares solution is given by*
$$\hat{x} = (A^T A)^{-1} A^T b.$$

Building a simple linear model with Python

The file `JSE_Car_Lab.csv`:

```
Price,Mileage,Make,Model,Trim,Type,Cylinder,Liter,Doors,Cruise,Sound,Leather
17314.1031289016,8221,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,1,1
17542.0360832793,9135,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,1,0
16218.8478619377,13196,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,1,0
16336.9131400486,16342,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,0,0
16339.1703239255,19832,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,0,1
15709.0528210833,22236,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,1,0
15230.0033898479,22576,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,1,0
15048.042184116,22964,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,1,0
14862.0938695978,24021,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,0,1
15295.0182668788,27325,Buick,Century,Sedan 4D,Sedan,6,3.1,4,1,1,1
```

Loading the data with the headers using Pandas:

```
import pandas as pd
data = pd.read_csv('./data/JSE_Car_Lab.csv',delimiter=',')
```

We extract the numerical columns:

```
y = np.array(data['Price'])
x = np.array(data['Mileage'])
x = x.reshape(len(x),1)
```


Building a simple linear model with Python (cont.)

The `scikit-learn` package provides a lot of very powerful functions/objects to analyse datasets.

Typical syntax:

1. Create object representing the model.
2. Call the `fit` method of the model with the data as arguments.
3. Use the `predict` method to make predictions.

```
from sklearn.linear_model import LinearRegression
lin_model = LinearRegression(fit_intercept=True)
lin_model.fit(x,y)

print lin_model.coef_
print lin_model.intercept_
```

We obtain $\text{price} \approx -0.17 \cdot \text{mileage} + 24764.5$.

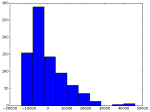
Measuring the fit of a linear model

How good is our linear model?

- We examine the *residual sum of squares*:
$$\mathrm{RSS}(\hat{\beta}) = \|y - X\hat{\beta}\|^2 = \sum_{k=1}^{n} (y_i - \hat{y}_i)^2.$$
`((y-lin_model.predict(x))**2).sum()`
We find: 76855792485.91. Quite a large error... The average absolute error:
`(abs(y-lin_model.predict(x))).mean()`
is 7596.28. Not so good...
- We examine the distribution of the residuals:
```
import matplotlib.pyplot as plt
plt.hist(y-lin_model.predict(x))
plt.show()
```

Measuring the fit of a linear model (cont.)

Histogram of the residuals:

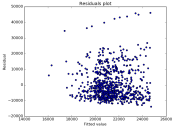


- Non-symmetric.
- Heavy tail.

- The heavy tail suggests there may be outliers.
- It also suggests transforming the response variable using a transformation such as $\log$, $\sqrt{\cdot}$, or $1/x$.

Measuring the fit of a linear model (cont.)

Plotting the residuals as a function of the fitted values, we immediately observe some patterns.

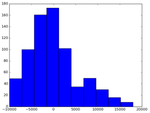


Outliers? Separate categories of cars?

Improving the model

- Add more variables to the model.
- Select the best variables to include.
- Use transformations.
- Separate cars into categories (e.g. exclude expansive cars).
- etc.

For example, let us use all the variables, and exclude Cadillacs from the dataset.



- Much more symmetric.
- Closer to a Gaussian distribution.

Average absolute error drops to 4241.21.