Regularization and Big Data

John Myles White

June 20, 2011

Data, data, data! I can't make bricks without clay!

According to reputable sources, we're in the Age of Big Data.

But what makes data big?

N

11	12	91	87
19	21	17	92
84	71	29	35
73	67	53	48

D

What happens as $N \to \infty$?

Traditionally, good things:

- Law of Large Numbers
- Central Limit Theorem
- Consistent estimators
- Asymptotic guarantees
- Large sample theory

What happens as $D \to \infty$?

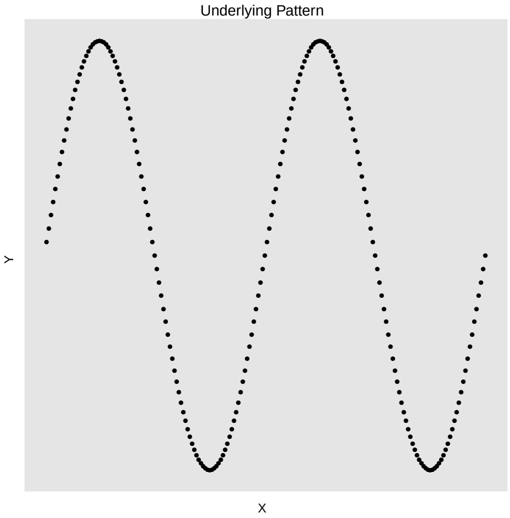
Traditionally, bad things:

- Underdetermined systems
- Infinitely many perfect models
- Massive overfitting

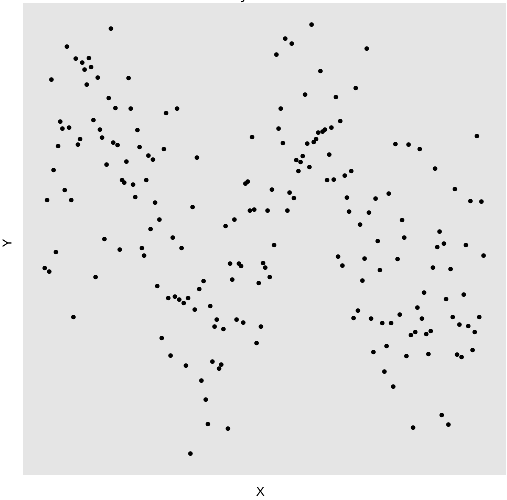
What leads to overfitting?

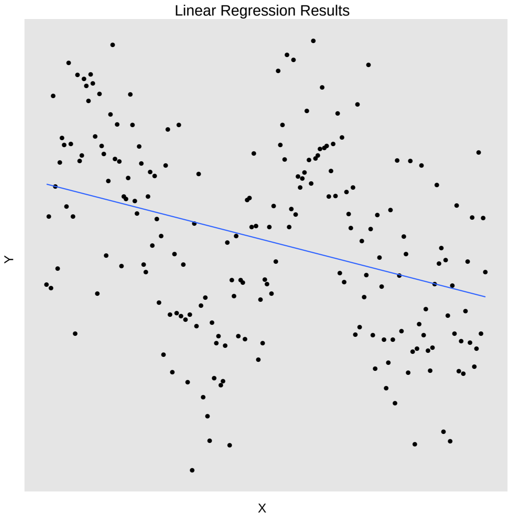
Example data:

- $N = 200$
- $D = 1$
- $y = sin(4\pi x) + \epsilon$
- $\epsilon \sim N(0, 0.75)$


Underlying Pattern
Y
X

Noisy Data Set




Linear Regression Results
Y
X

Linear regression:

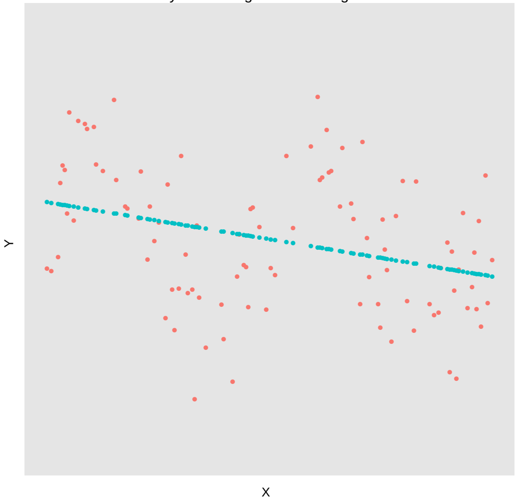
- $Y = \beta_0 + \beta_1 X$

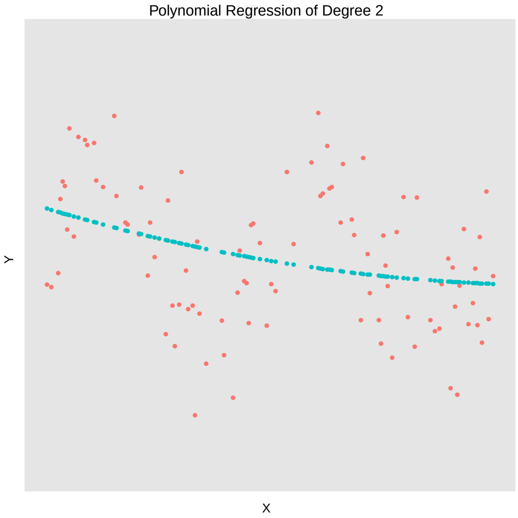
Linear regression isn't very expressive. We need a stronger model.

Polynomial regression:

- $Y = \beta_0 + \beta_1 X + \beta_2 X^2$
- $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3$
- $\ldots$
- $Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \ldots + \beta_{20} X^{20}$

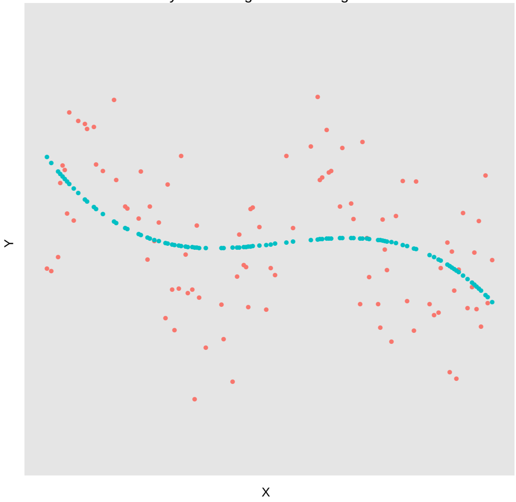
Polynomial Regression of Degree 1

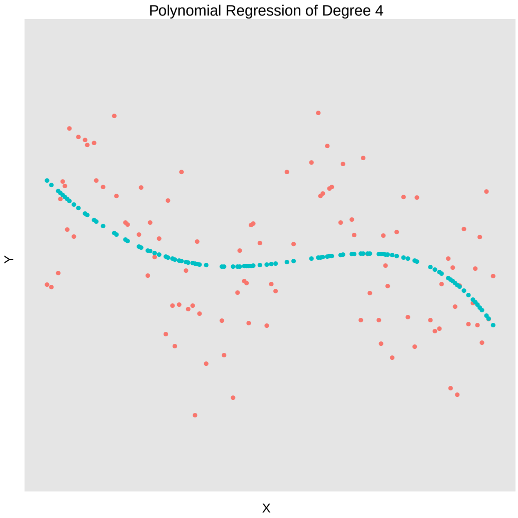




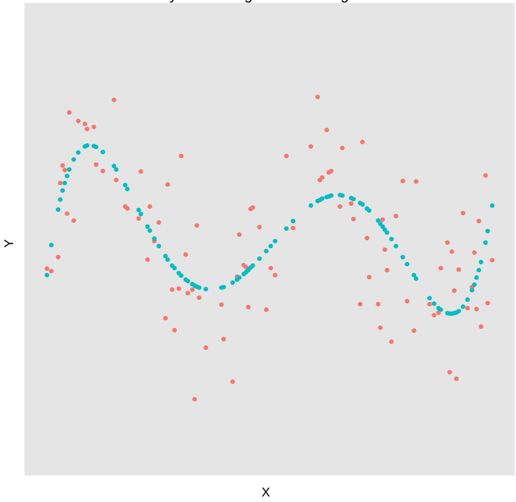
Polynomial Regression of Degree 2
Y
X

Polynomial Regression of Degree 3

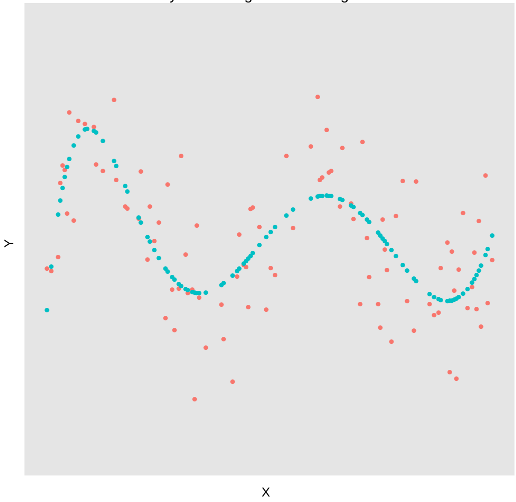


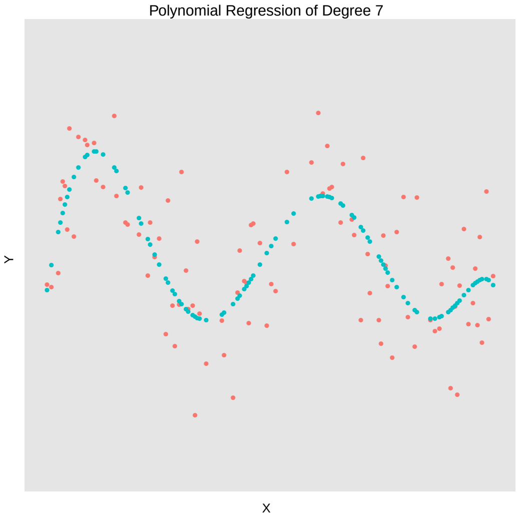

Polynomial Regression of Degree 4
Y
X

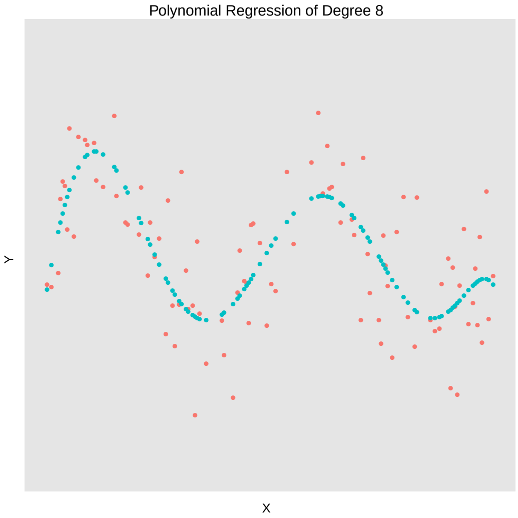
Polynomial Regression of Degree 5



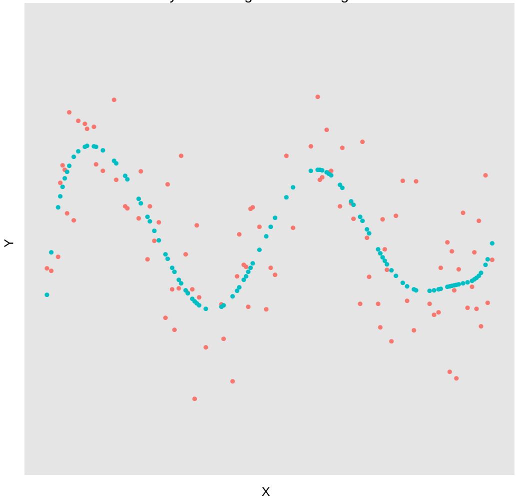
Polynomial Regression of Degree 6



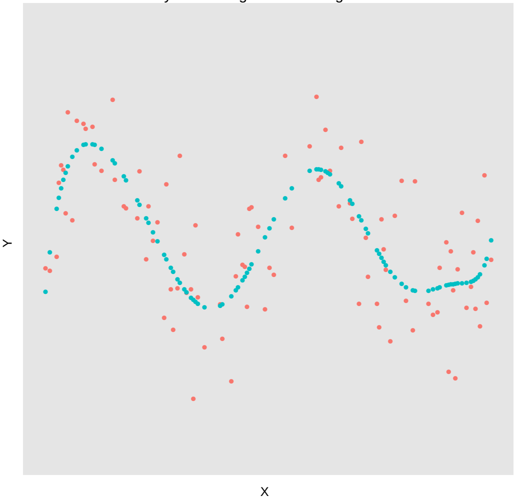

Polynomial Regression of Degree 7
Y
X

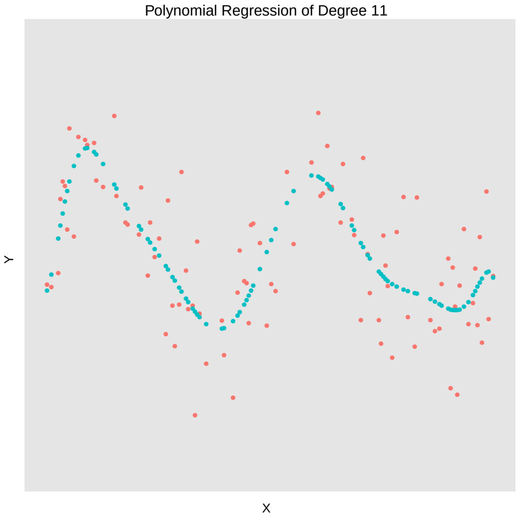

Polynomial Regression of Degree 8
Y
X

Polynomial Regression of Degree 9

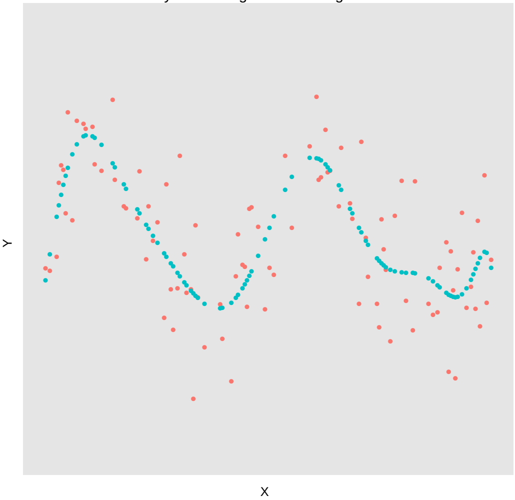


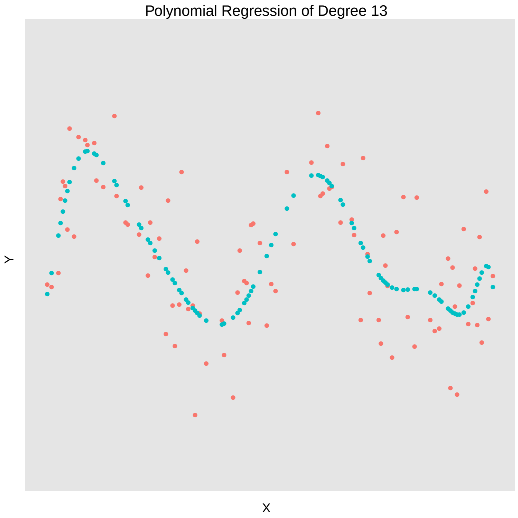
Polynomial Regression of Degree 10



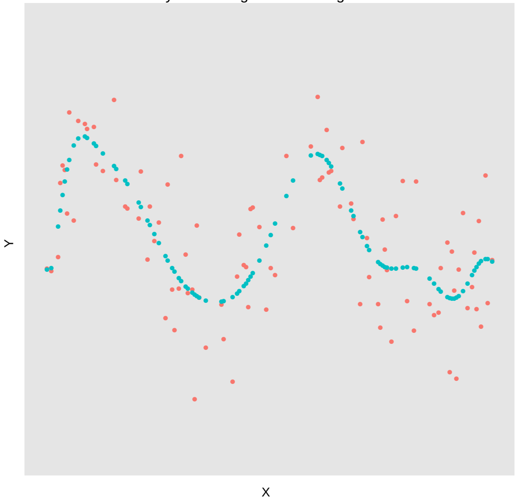

Polynomial Regression of Degree 11
Y
X

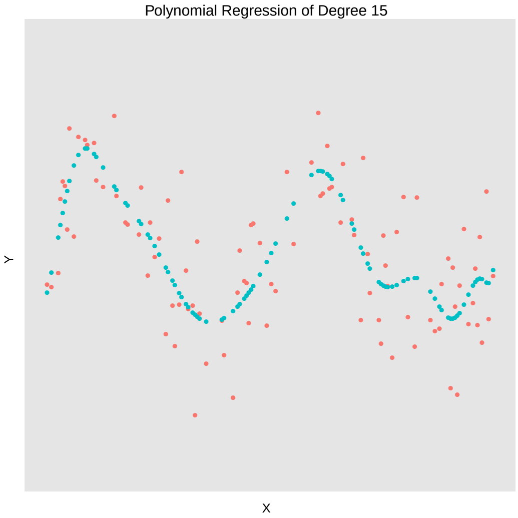
Polynomial Regression of Degree 12




Polynomial Regression of Degree 13
Y
X

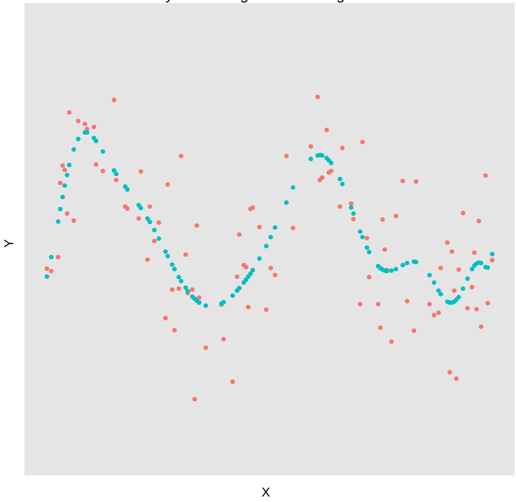
Polynomial Regression of Degree 14



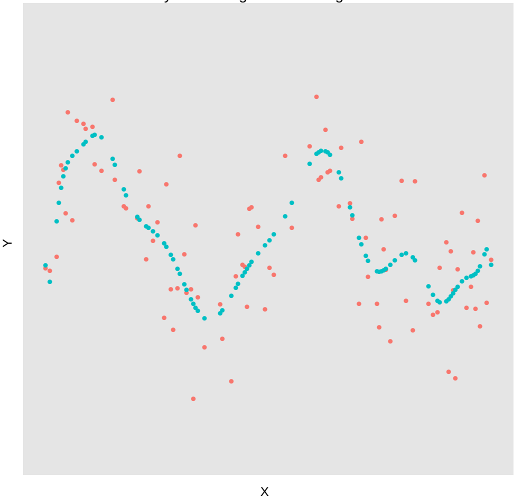


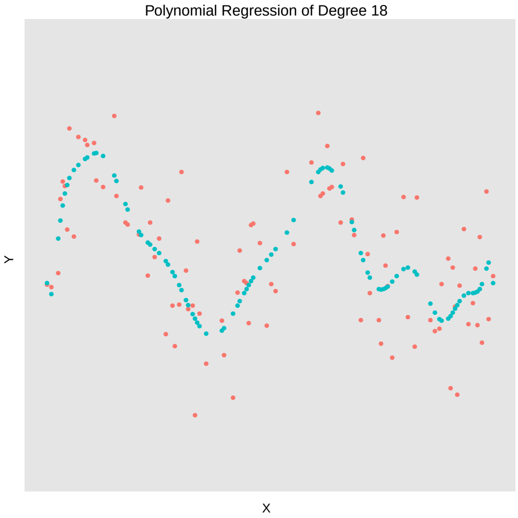
Polynomial Regression of Degree 15
Y
X

Polynomial Regression of Degree 16

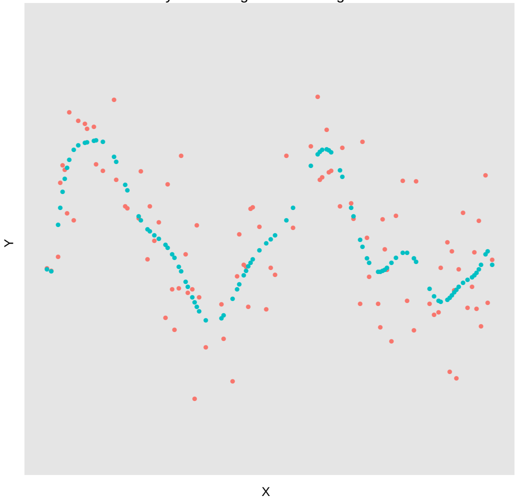


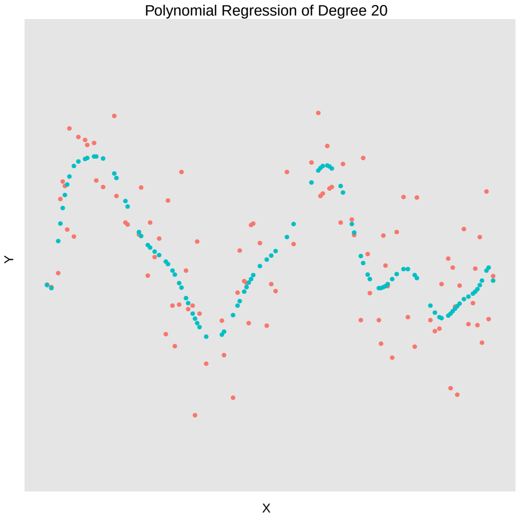
Polynomial Regression of Degree 17



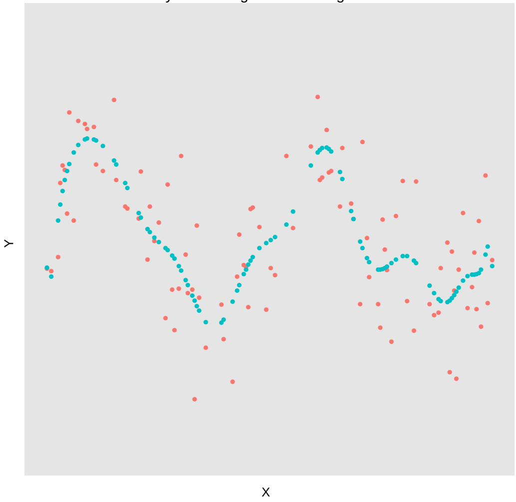

Polynomial Regression of Degree 18
Y
X

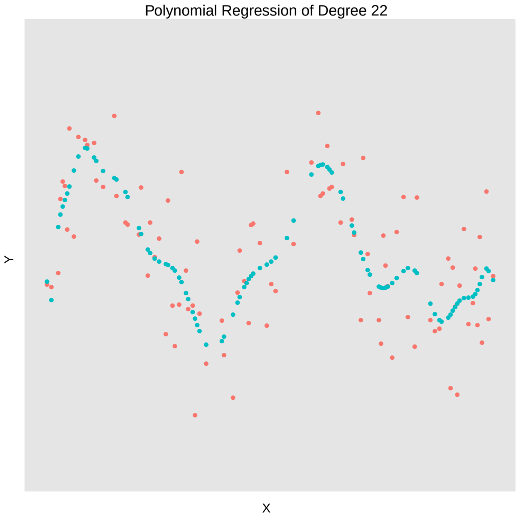
Polynomial Regression of Degree 19




Polynomial Regression of Degree 20
Y
X

Polynomial Regression of Degree 21

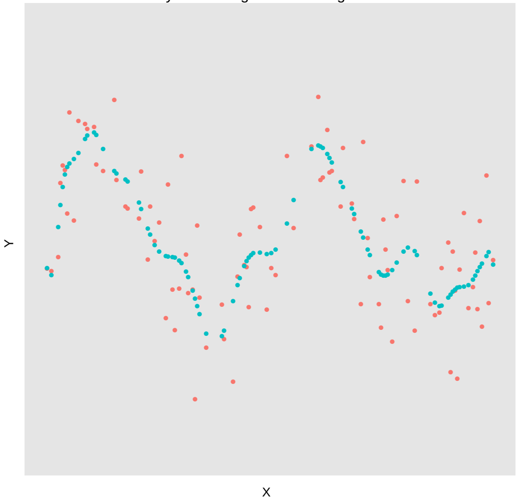



Polynomial Regression of Degree 22
Y
X

Polynomial Regression of Degree 23
Y
X


Polynomial Regression of Degree 24
Y
X

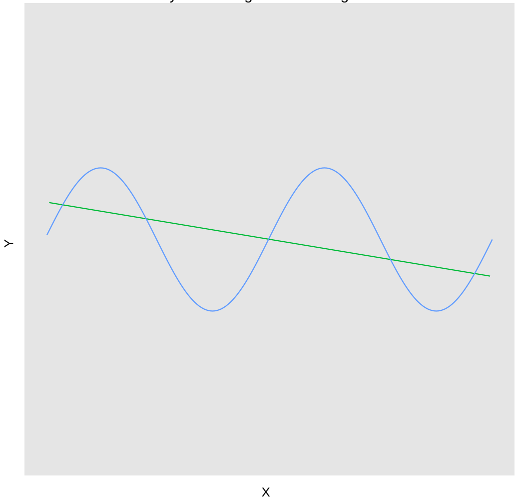
Polynomial Regression of Degree 25



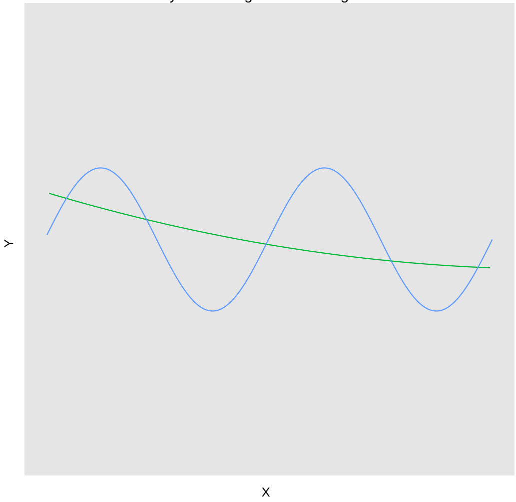
As $D \rightarrow N$, we fit the data better and better.

But our model gets further and further from the true pattern…

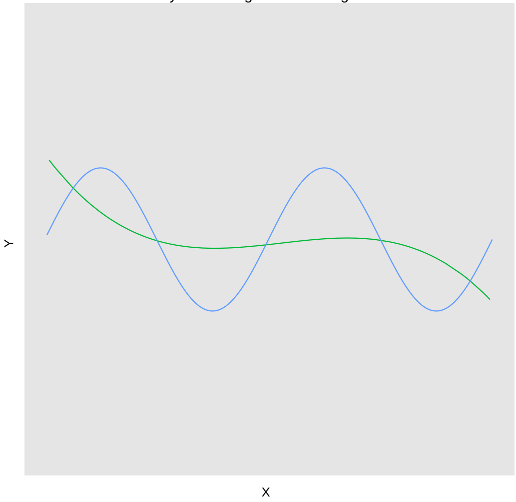
Polynomial Regression of Degree 1



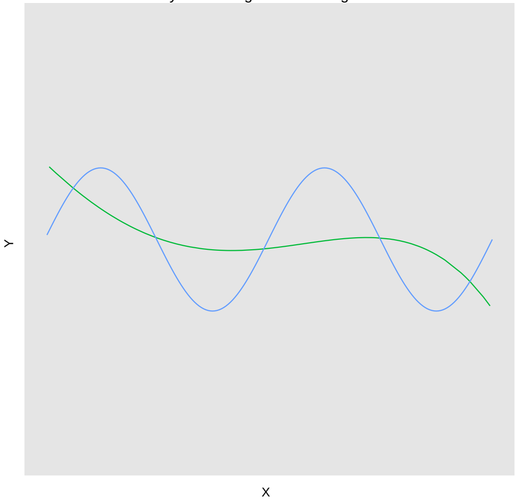
Polynomial Regression of Degree 2

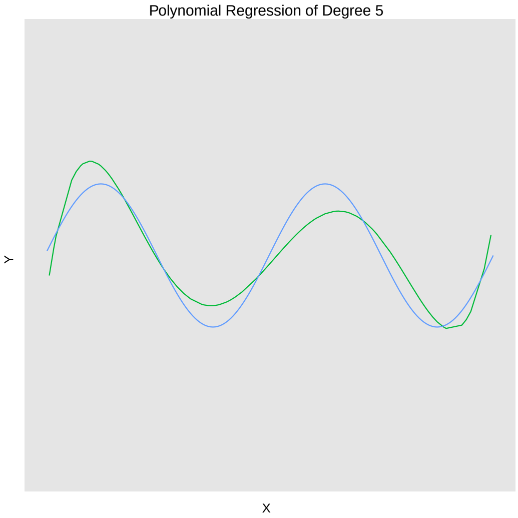


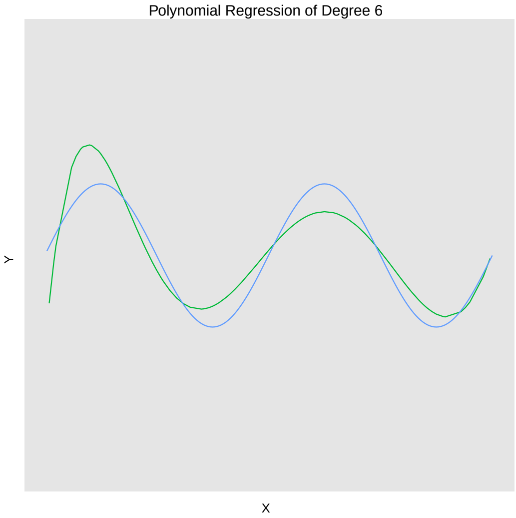
Polynomial Regression of Degree 3



Polynomial Regression of Degree 4

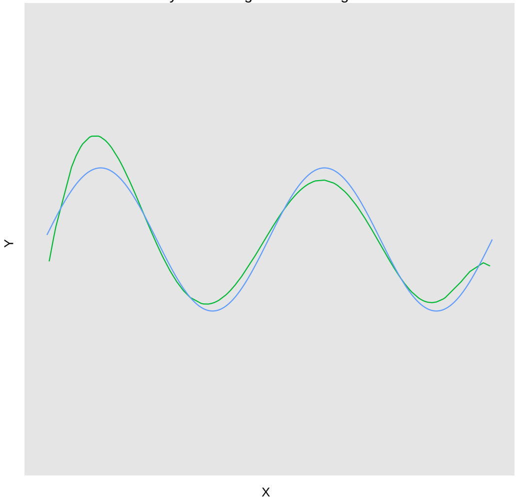



Polynomial Regression of Degree 5
Y
X

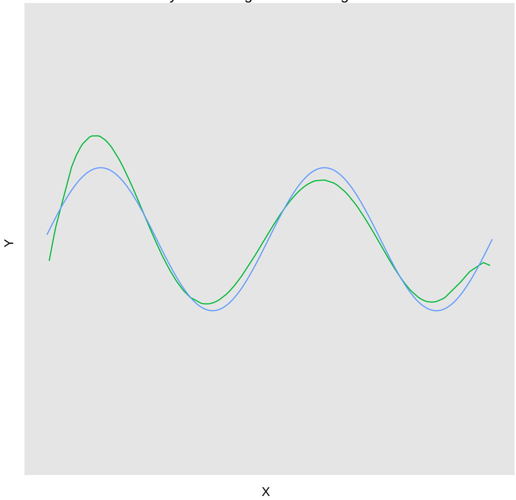


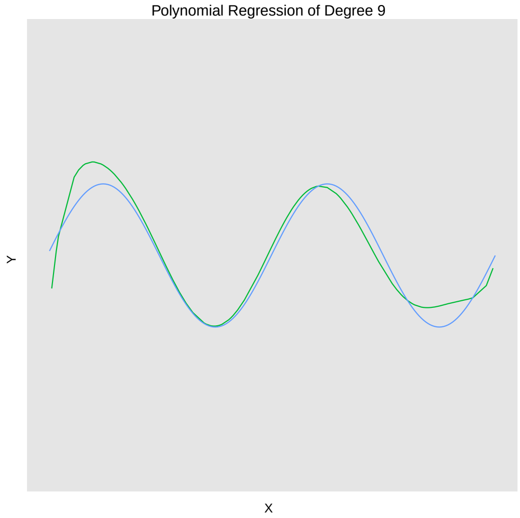
Polynomial Regression of Degree 6
Y
X

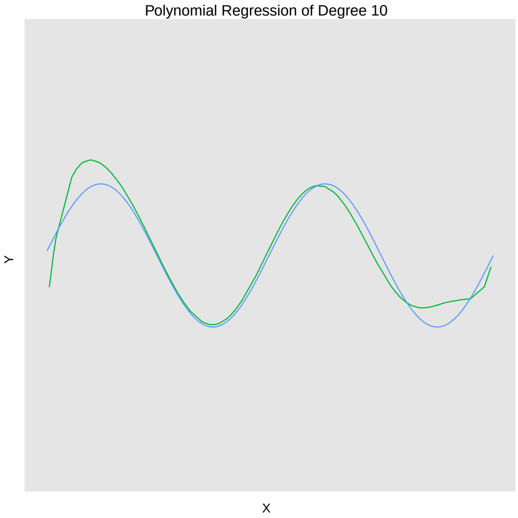
Polynomial Regression of Degree 7



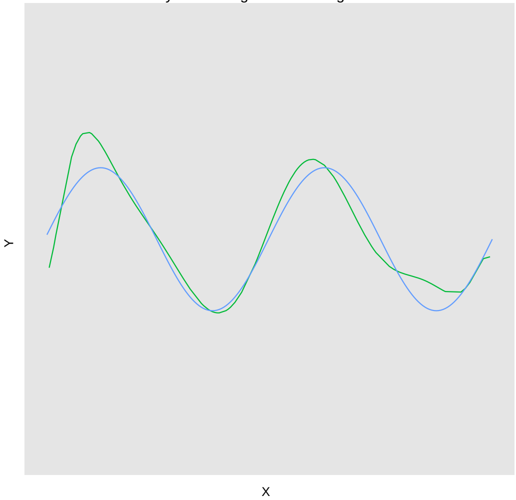
Polynomial Regression of Degree 8

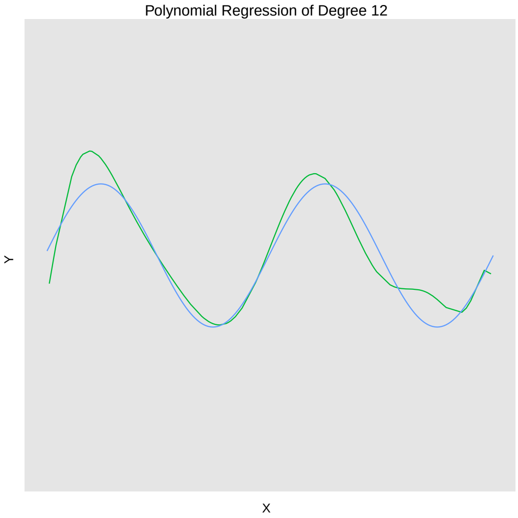



Polynomial Regression of Degree 9
Y
X

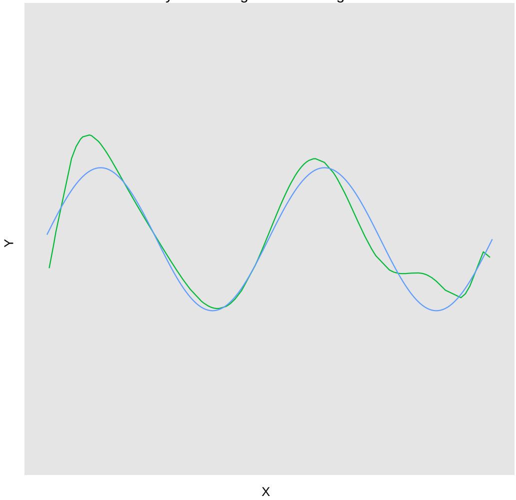

Polynomial Regression of Degree 10
Y
X

Polynomial Regression of Degree 11

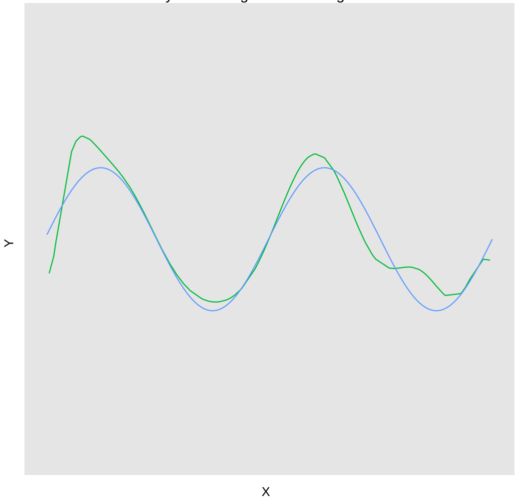


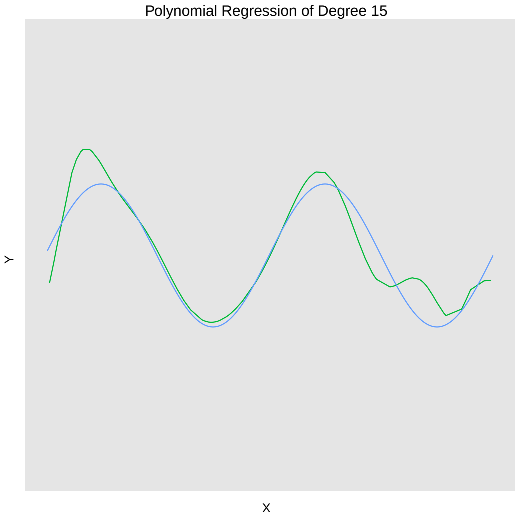

Polynomial Regression of Degree 12
Y
X

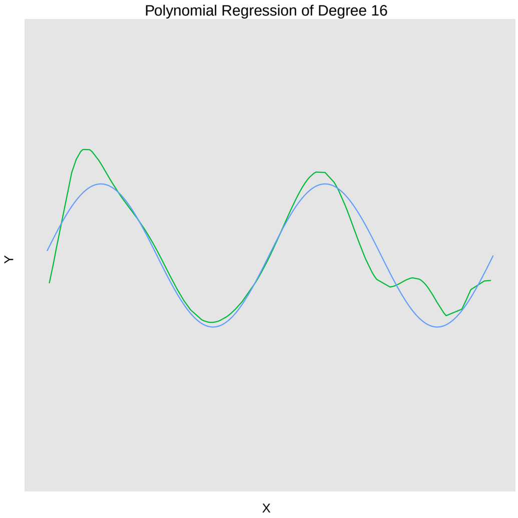
Polynomial Regression of Degree 13



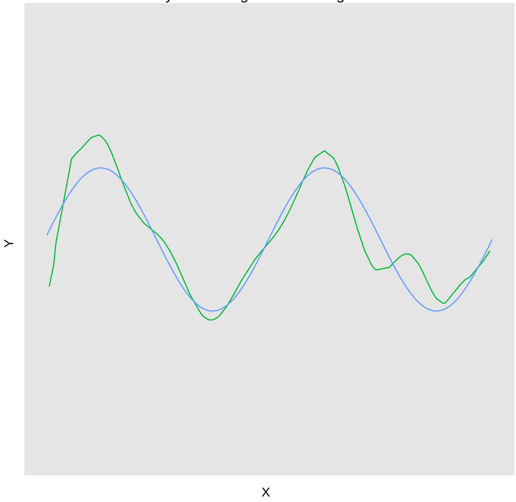
Polynomial Regression of Degree 14



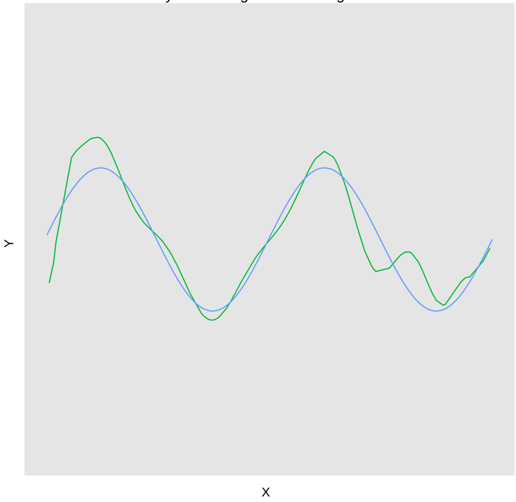

Polynomial Regression of Degree 15
Y
X


Polynomial Regression of Degree 16
Y
X

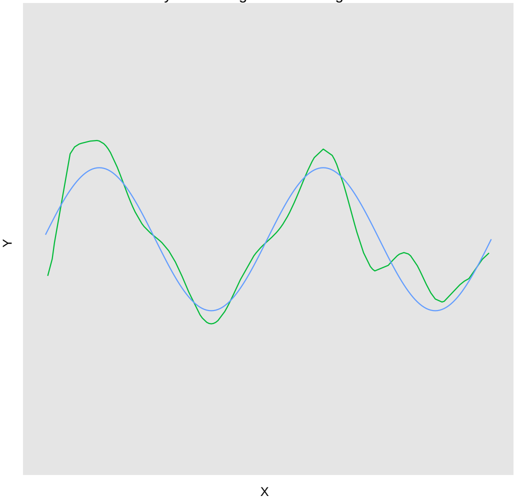
Polynomial Regression of Degree 17

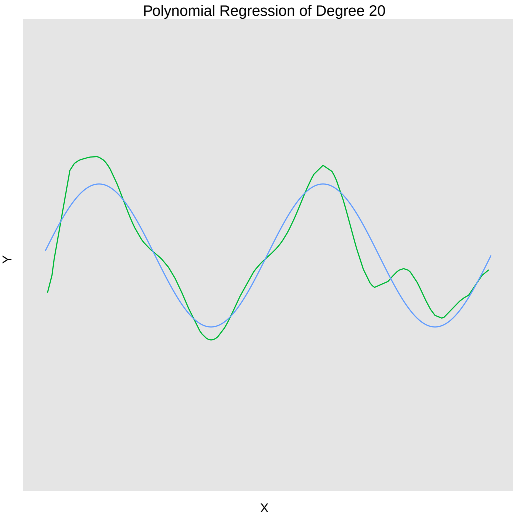


Polynomial Regression of Degree 18

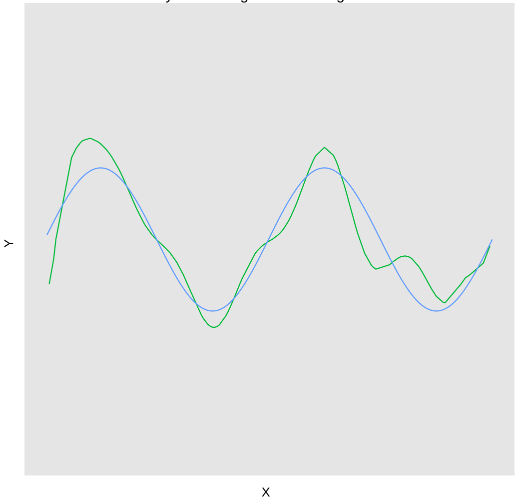


Polynomial Regression of Degree 19

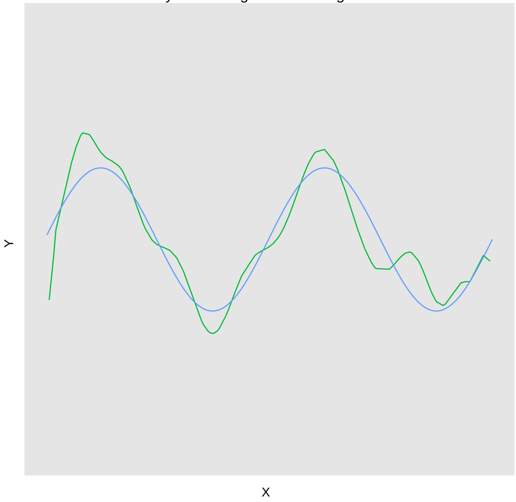


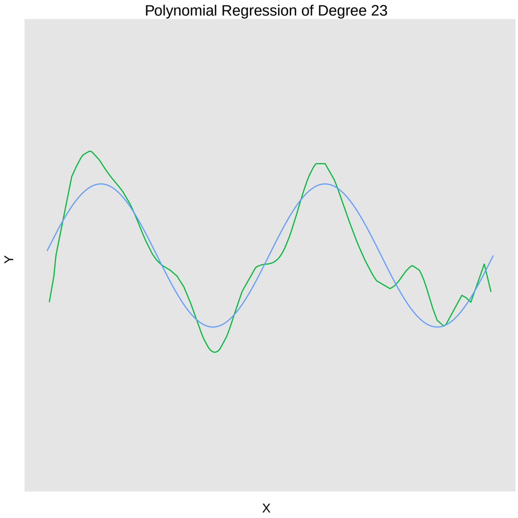

Polynomial Regression of Degree 20
Y
X

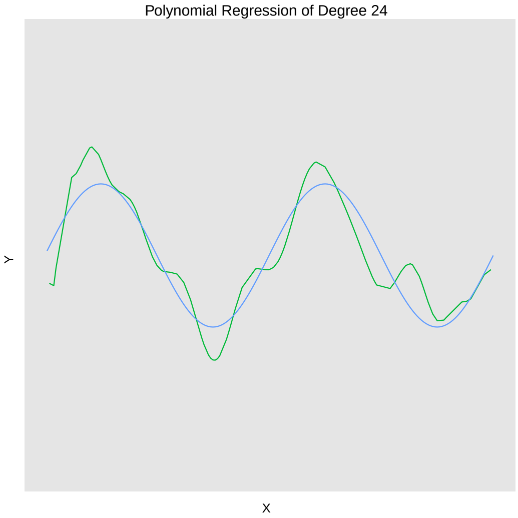
Polynomial Regression of Degree 21



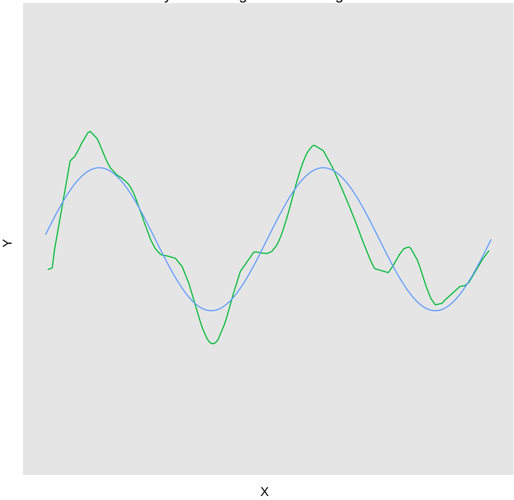
Polynomial Regression of Degree 22




Polynomial Regression of Degree 23
Y
X

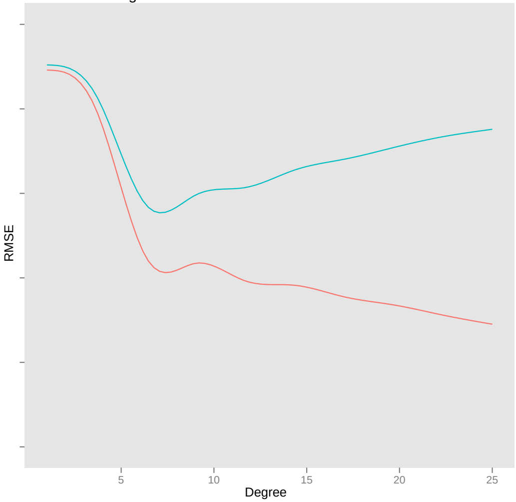

Polynomial Regression of Degree 24
Y
X

Polynomial Regression of Degree 25



Overfitting always occur when our models become too expressive

Training Set Performance versus Test Set Performance

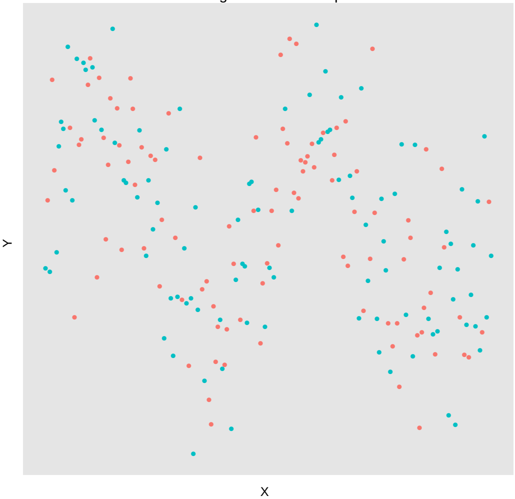


How do we prevent overfitting?

One approach:

- Split data into training set and test set
- Pick model that does best on held out test set

Training Set / Test Set Split



This underestimates the strongest model we can safely use.

Another approach:

- Regularize our model

Unregularized models minimize prediction error:

- OLS regression
- Logistic regresion

$$\beta^* = \arg\min_{\beta} (Y - X\beta)^2$$

Regularized models minimize prediction error and model size:

- Ridge regression
- Lasso regression

$$\beta^* = \arg\min_{\beta} (Y - X\beta)^2 + \beta^2$$

$$\beta^* = \arg\min_\beta (Y - X\beta)^2 + |\beta|$$

How can we use regularization to solve real world problems?

The text regression problem:

- N documents
- D words
- Predict continuous value for each document from word counts

Examples:

- From IPO notices, predict stock volatility
- From press releases, predict Congress member's politics
- ...

We need regularization because we:

- Observe more words than documents
- Want sparse solutions, e.g. a few words that matter a lot

From press releases, how can we find words that signal politics?

Hey, does anybody notice this crazy thing that we're on the road to socialism? I'm just saying. Wow. We got — we got the SCHIPs thing going for us. That's great.

How about that McDonalds two blocks from Ground Zero? That's killed more people than the nineteen hijackers.

Who thinks:

1. Text A was pro-Democrat and Text B was pro-Republican?
2. Text A was pro-Republican and Text A was pro-Democrat?

Corpus Statistics:

- $N = 1,408$ unique documents
- $D = 20,521$ unique words

Document A:

i want to talk about jobs lately it seems that everyone says they want to talk about jobs and that we'll get around to tackling jobs next week or the week after

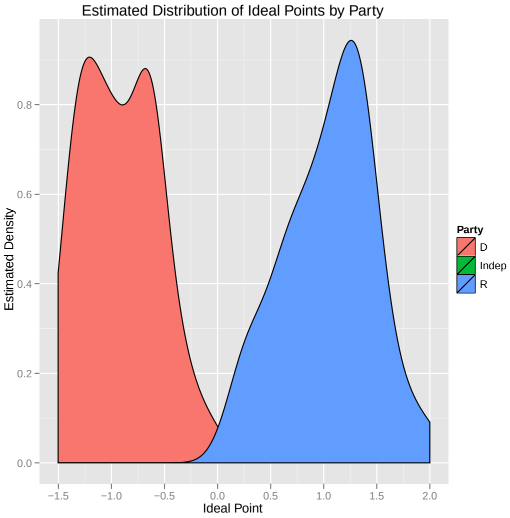
Document B:

there was a major legislative accomplishment in washington last week and it's getting less attention than it deserves because it isn't national health care reform

Document Term Matrix:

Document	I	Want	Talk	Jobs	Week
A	1	2	2	3	2
B	0	0	0	0	1

- Fit Lasso regression to word counts
- Predict ideal points for senators


Estimated Distribution of Ideal Points by Party
Estimated Density
0.0
0.2
0.4
0.6
0.8
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
Ideal Point
Party
D
Indep
R

Top 10 Most Republican Terms:

Term	Value
okla	1.23
bailey	0.647
johnny	0.588
administering	0.561
neb	0.556
sam	0.542
986	0.532
texans	0.493
patriotism	0.466
demint	0.417

Top 10 Most Democratic Terms:

Term	Value
sherrod	-0.367
sheldon	-0.249
dec	-0.196
possess	-0.168
salaries	-0.158
tom	-0.152
debbie	-0.151
dark	-0.148
lautenberg	-0.133
fought	-0.106

Debugging:

- Too many names of senators in our list
- Strip out all the names from corpus
- Run analysis from scratch on clean corpus

Top 10 Most Republican Terms excluding Names:

Term	Value
okla	1.13
neb	0.726
bailey	0.674
2415	0.638
986	0.578
kansans	0.543
administering	0.516
texans	0.467
profoundly	0.459
patriotism	0.430

Top 10 Most Democratic Terms excluding Names:

Term	Value
cedar	-0.224
chaired	-0.197
dec	-0.158
dark	-0.146
blocked	-0.138
reverses	-0.134
1960s	-0.125
insurers	-0.0958
fought	-0.0926
possess	-0.0923