LECTURE 16:

BEYOND LINEARITY PT. 1

November 6, 2017

SDS 293: Machine Learning

Announcements

- Assignments
  - Feedback for A5/A6 should be out shortly
  - Solutions posted to Moodle
- Final Projects:
  - FP1: Data Appendix due Wednesday*
  - 6 people still need teams
- A few minor schedule changes pending
  - Mostly in response to final project topics
  - Will announce on Slack when confirmed
- T-minus **6 weeks** until the end of the semester!

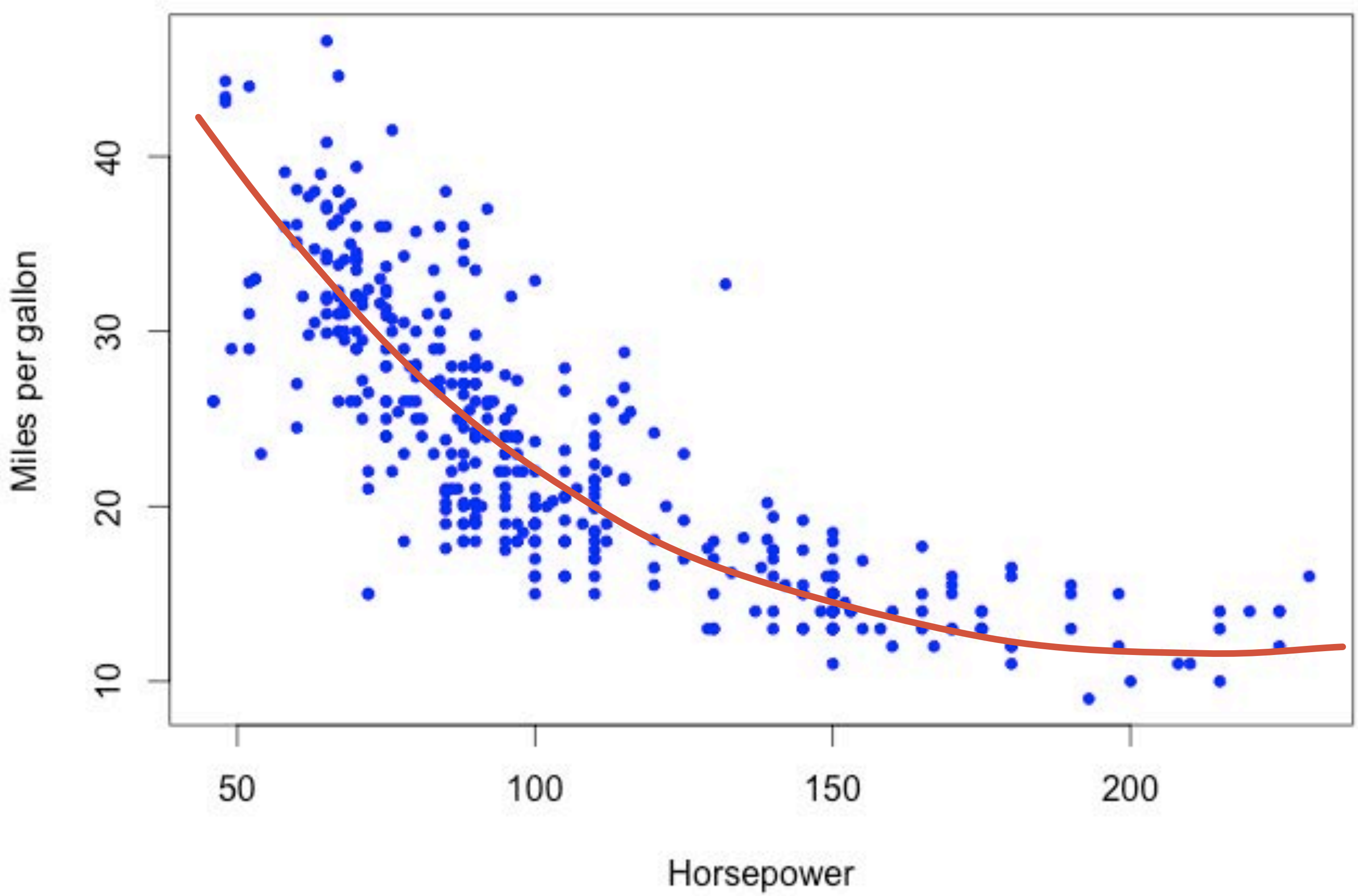
Outline

- Moving beyond linearity
  - Polynomial regression
  - Step functions
  - Splines
  - Local regression
  - Generalized additive models (GAMs)
- Labs for each part

So far: linear models

- The good:
  - Easy to describe & implement
  - Straightforward interpretation & inference
- The bad:
  - Linearity assumption is (almost) always an approximation
  - Sometimes it's a pretty poor one
- RR, the lasso, PCA, etc. all try to improve on least squares by **controlling the variance** of a linear model
- … but linear models can only **stretch so far**

Flashback: `Auto` dataset



Polynomial regression

- One simple fix is to use **polynomial transformations:**

$$\texttt{mpg} = \beta_0 + \beta_1 \times \texttt{horsepower} + \beta_2 \times \texttt{horsepower}^2 + \epsilon$$

- This example is a *quadratic regression*

- **Big idea:** extend the linear model by adding extra predictors that are powers of the original predictors

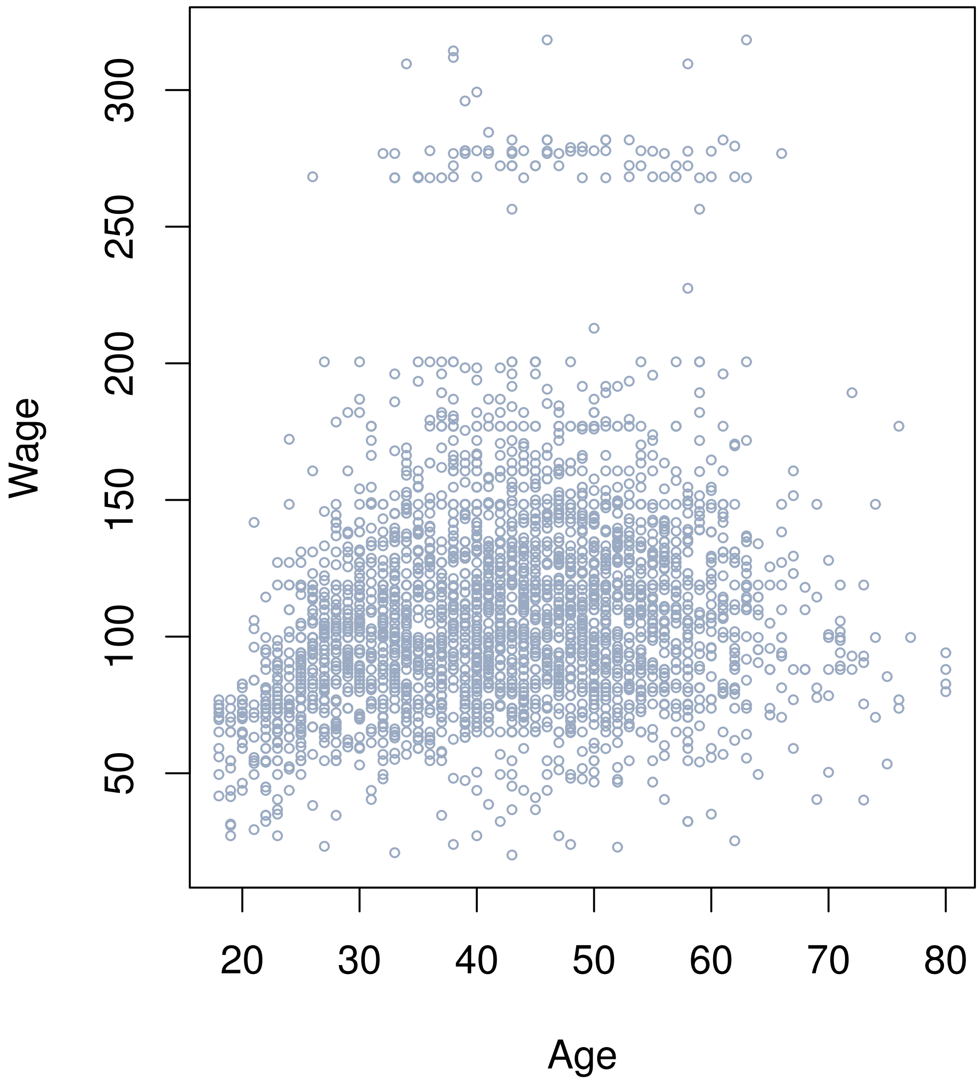
Note: this is still a linear model!

(and so we can find its coefficients using regular ol' least squares)

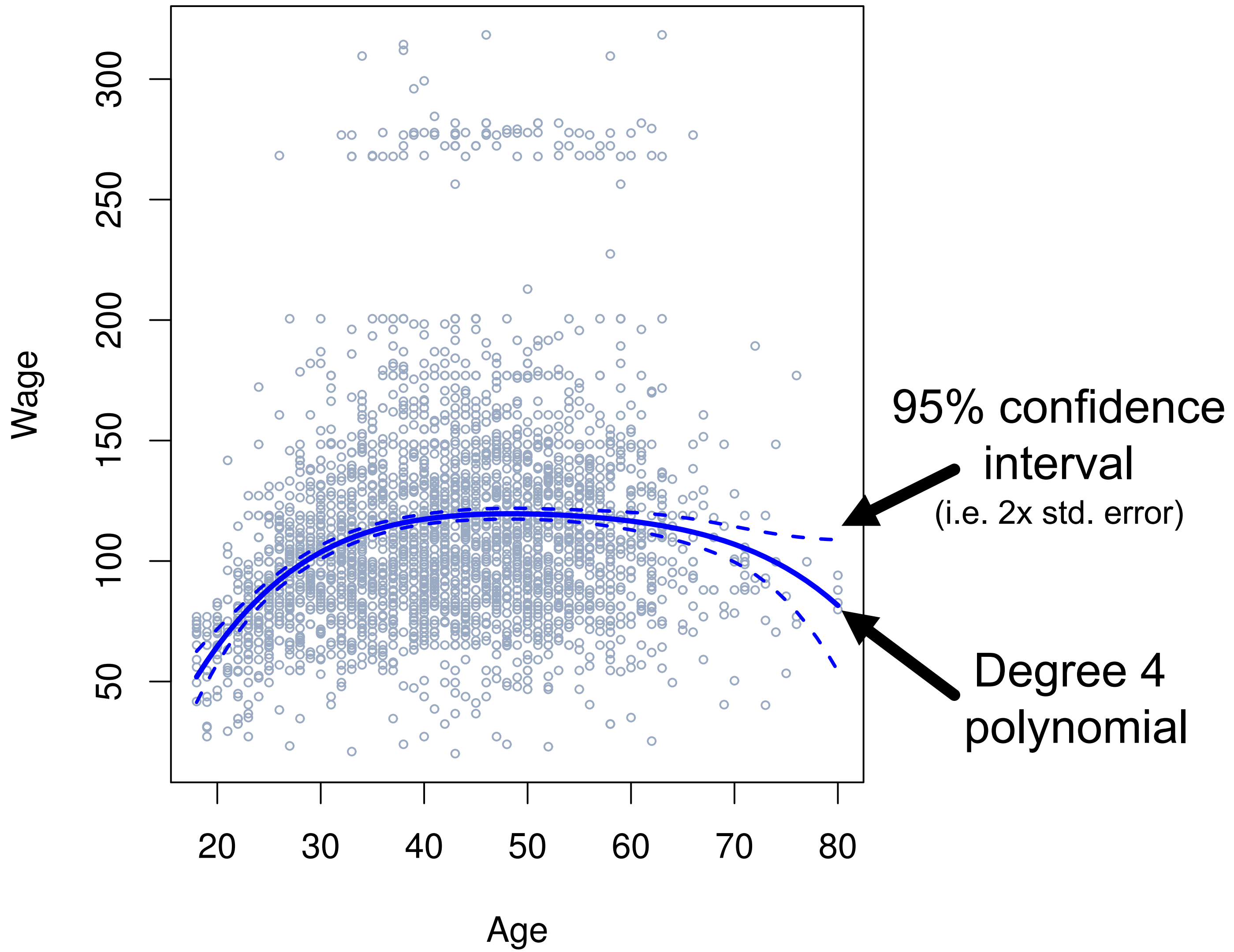
Polynomial regression in practice

- For large enough degree d, a polynomial regression allows us to produce an extremely non-linear curve
- As d increases, this can produce some really weird shapes
- **Question**: what's happening in terms of bias vs. variance?
- **Answer**: increased flexibility → less bias, more variance; in practice, we generally only go to degree 3 or 4 unless we have additional knowledge that more will help

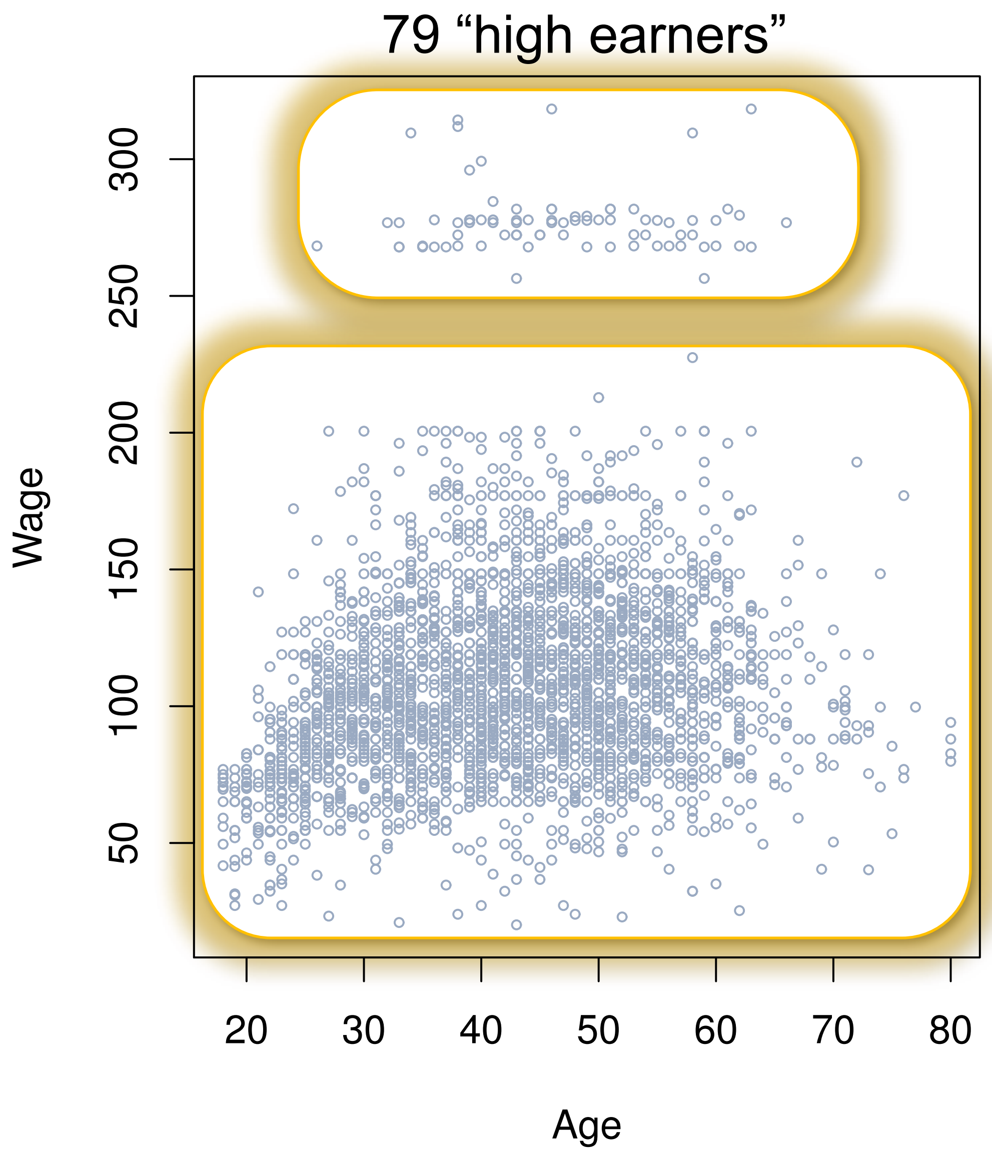
Example: `Wage` dataset



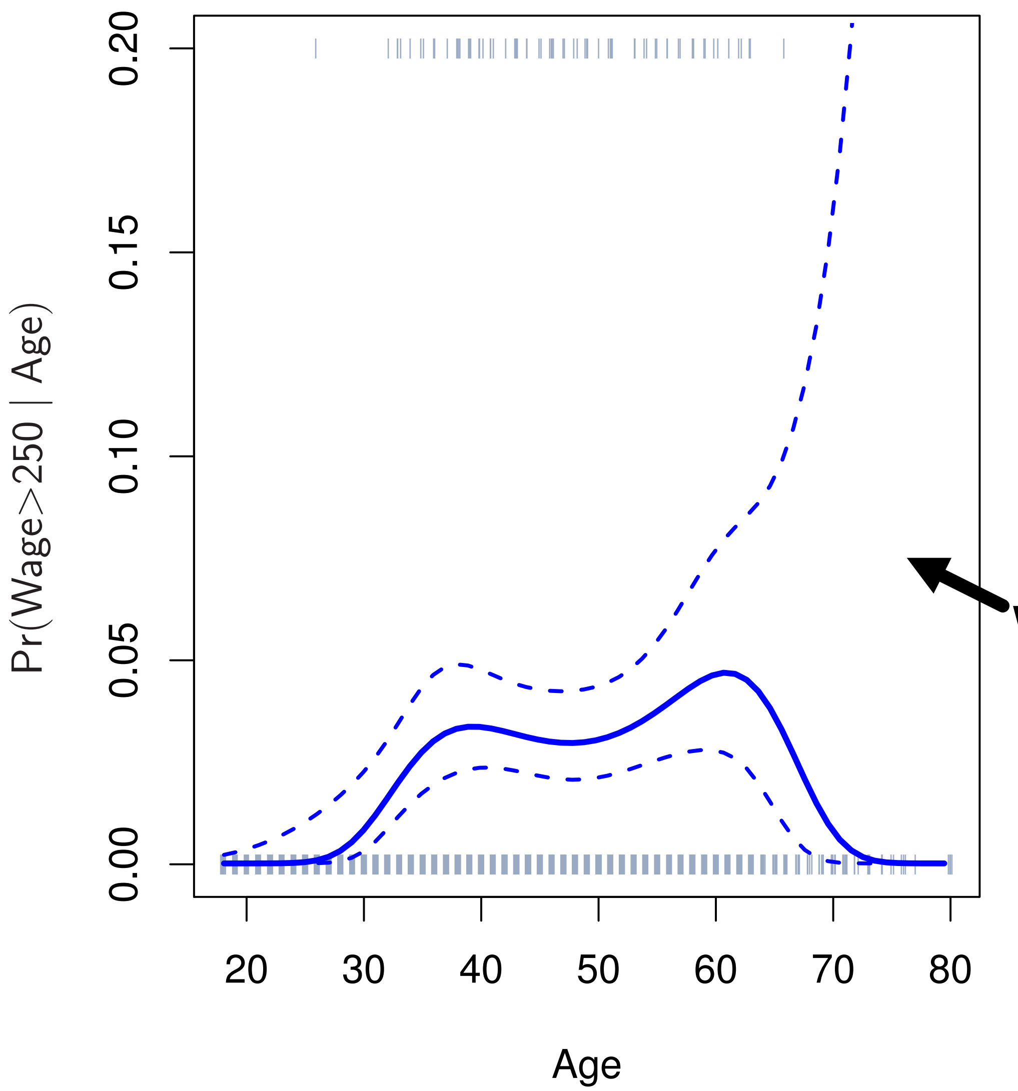
Example: `Wage` dataset



Example: `Wage` dataset

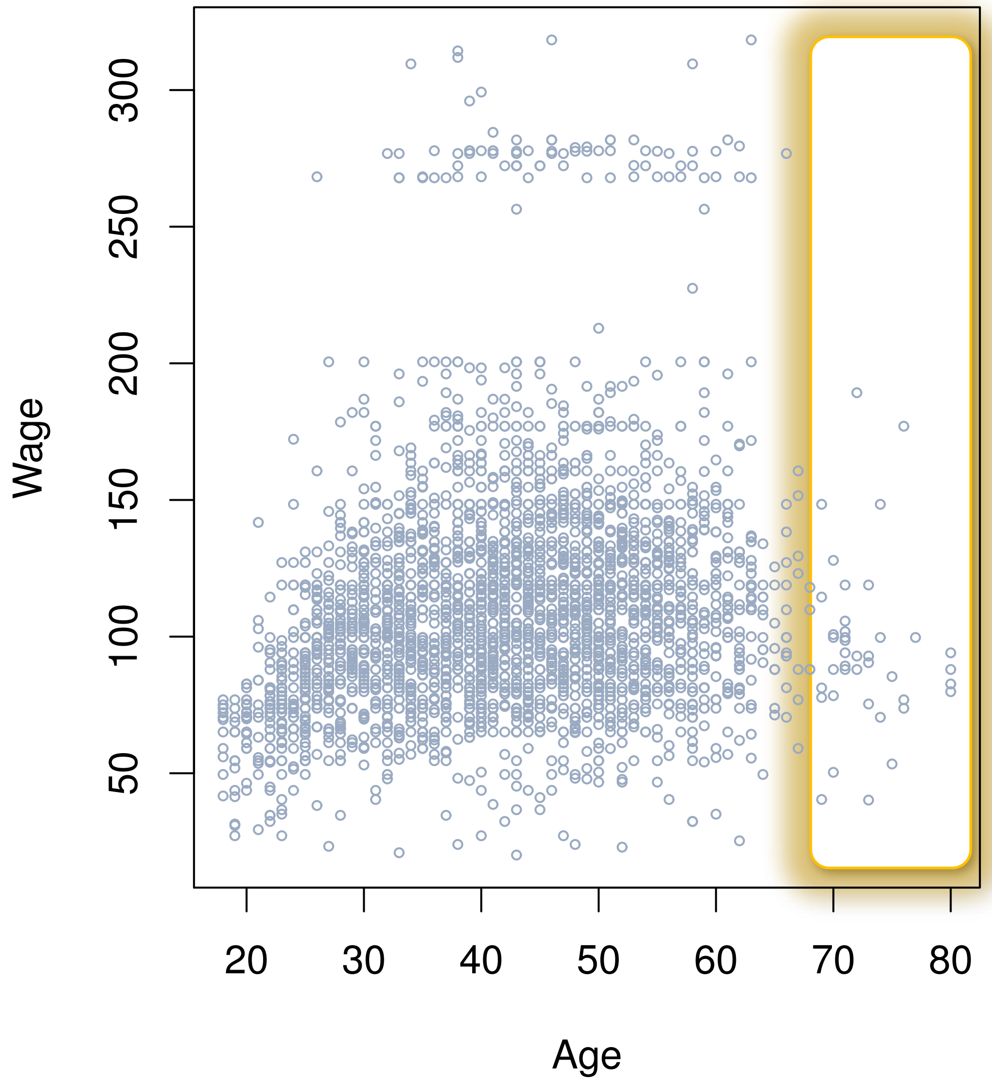


Example: `Wage` dataset



What's going on here?

Example: Wage dataset



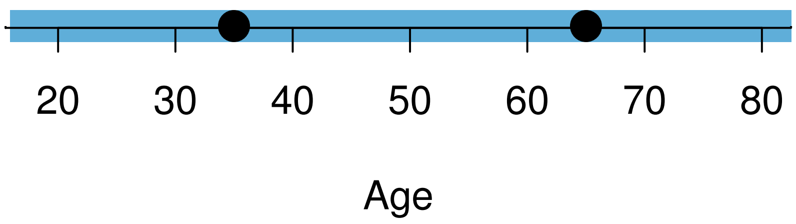
Relatively sparse
= less confident

Global structure in polynomial regression

- Polynomial regression gives us added flexibility, but imposes **global structure** on the non-linear function of X
- **Question**: what's the problem with this?
- **Answer**: when data behave differently in different parts of the domain, function can to get really complicated

Step functions

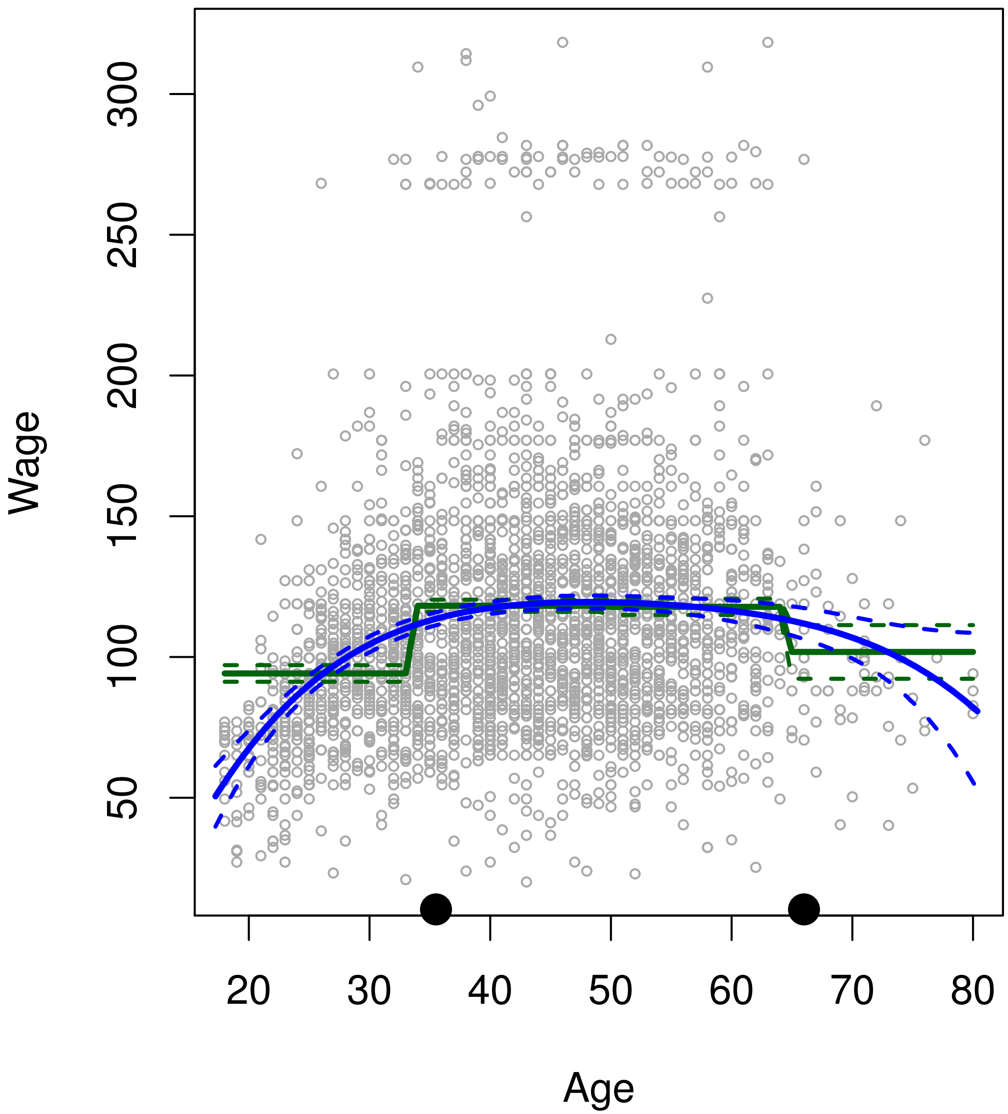
- **Big idea**: if our data exhibits different behavior in different parts, we can fit a separate “mini-model” on each piece and then glue them together to describe the whole
- Process:
  1. Create k cutpoints $c_1, c_2, \ldots, c_K$ in the range of X
  2. Construct (k+1) dummy variables:



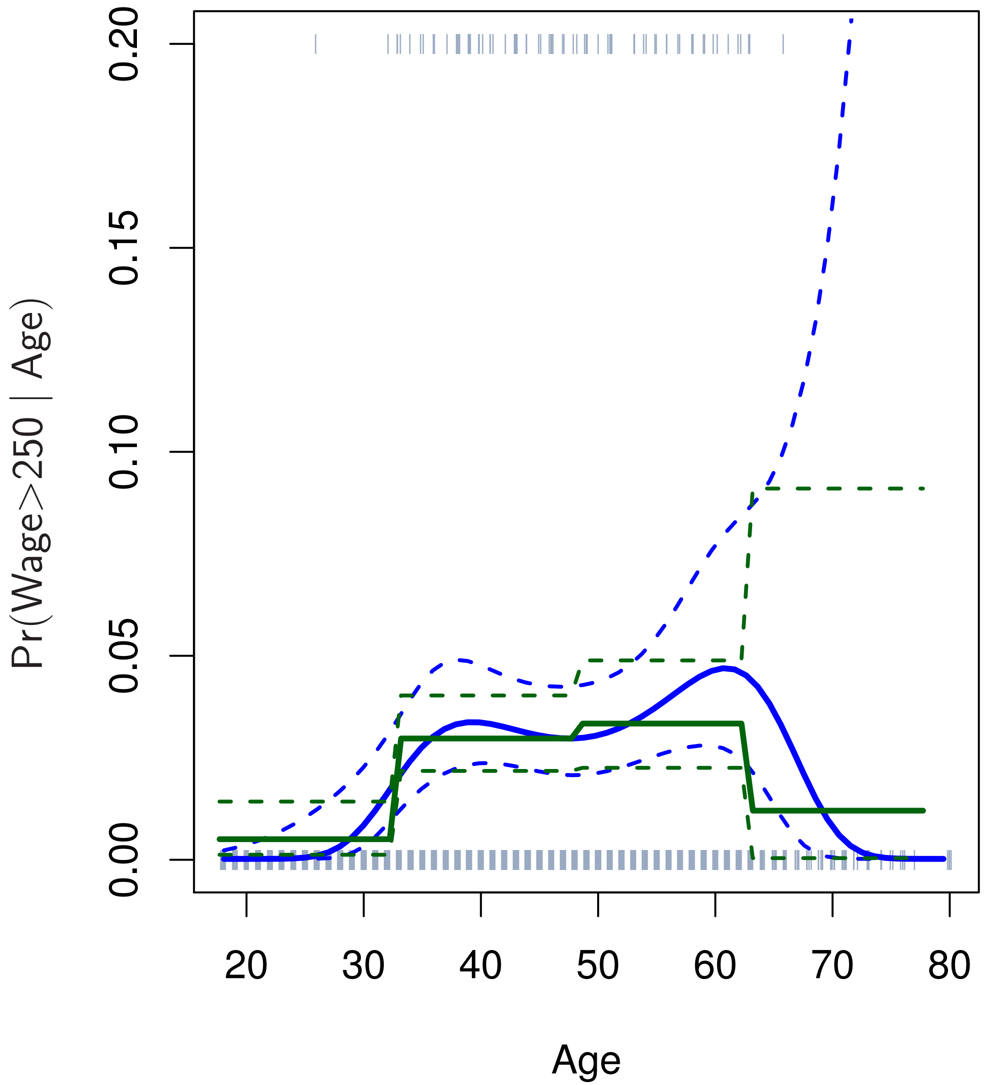
$$
\begin{aligned}
C_0(X) &= I(X < c_1) \\
C_1(X) &= I(c_1 \leq X < c_2) \\
&\vdots \\
C_K(X) &= I(c_k \leq X)
\end{aligned}
$$

  3. Fit least squares model using $C_1(X), \ldots, C_K(X)$ as predictors (we can exclude $C_0(X)$ because it is redundant with the intercept)

Example: `Wage` dataset



Example: `Wage` dataset



Granularity in step functions

- Step functions give us added flexibility by letting us model different parts of X independently
- **Question**: what's the problem with this?
- **Answer**: if our data doesn't have natural breaks, choosing the wrong step size might mean that we "miss the action"

Lab: Polynomials and Step Functions

- To do today’s lab in R: <nothing new>
- To do today’s lab in python: <nothing new>
- Instructions and code:

  http://www.science.smith.edu/~jcrouser/SDS293/labs/lab12/
- Full version can be found beginning on p. 287 of ISLR