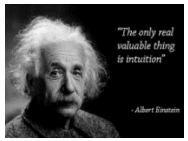
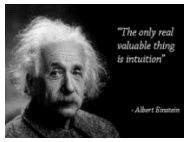
Step Models



Step Functions: Intuition

- **Polynomials** impose a mathematical **structure** in the model
- Complex mathematical structures generally provide **good fit** in the **training** data but often perform **poorly** with **test** data, especially in the tail ends (**"wagging the tail"**).
- If we note patterns in scatterplots showing that relationships **shift** at various **ranges** of the data, a step function breaking up the data into such sections may provide better predictions
- Step functions are pretty simple, the function predicts a **mean value** for **each section**
- For **example**, if predicting **wage** as a function of **age**, perhaps you notice that wages are generally higher after the age of 35, but then they tend to be lower after the age of 65. A step function will predict an average wage for **age <35**, another for **35 <= age < 65** and another for **age >= 65**



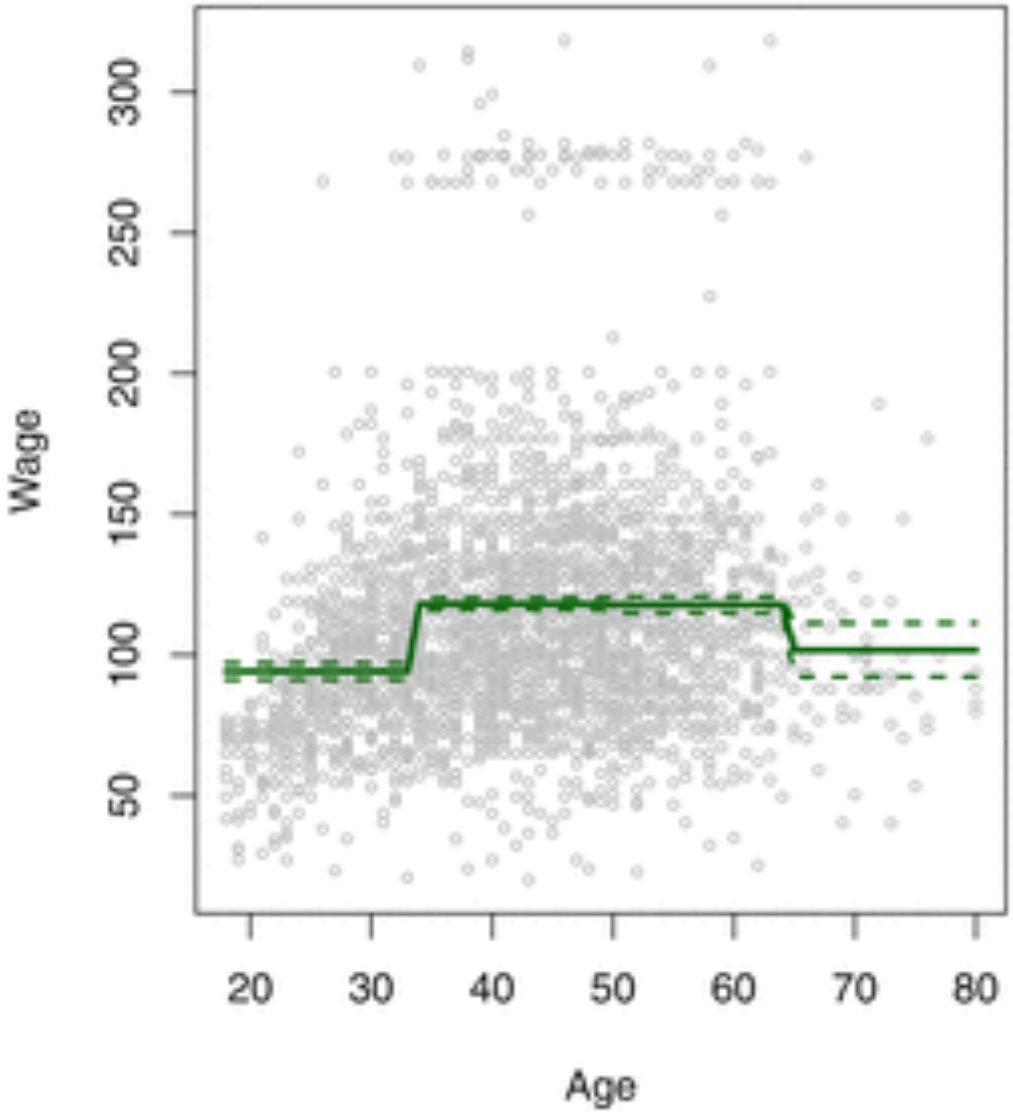
Step Function Illustration

Example:

Wage = $95K
for Age < 35

Wage = $150K
for 35 <= Age < 65

Wage = $105K
for Age >= 65



Tips

`cut()` → Function to partition the data for step functions

`fit.step=lm(y~cut(x,4),data=dataName)` → Cuts `x` into `4` equal segments

KOGOD SCHOOL
of
BUSINESS