Intro to Data Analytics and Visualizations

Lecture 11 – Managing Data
Fall 2014, September19

Outline

1. Fixing data quality problems
    1.1 Dealing with Missing Values
    1.2 Transformations
2. **Organizing your data for the modeling process**

Why Manage?

- Spotted issues in the exploring step (numerical summaries and visualizations)
- Decide what to do about the spotted issues
- Fix
- Re-Organize

Note: keep track of everything you do.

Sampling and Validation

- Test and training splits
- Creating a sample group column
- Record grouping
- Data provenance

Sampling and Validation

- Sample to:
- Make models run faster;
- Make graphs more informative;
- Ensure that the dataset is representative;

- Similar to population sampling for political polling.

Test and Training Splits

- Training data set: available data that you use to build a model;
- Test (or hold-out) set: once the model is built, does it work correctly? Us the test set to “test” the model.
- Split the available dataset in two parts: training and test set before you start modeling. Next insurance example could use that.

Data Science Problem

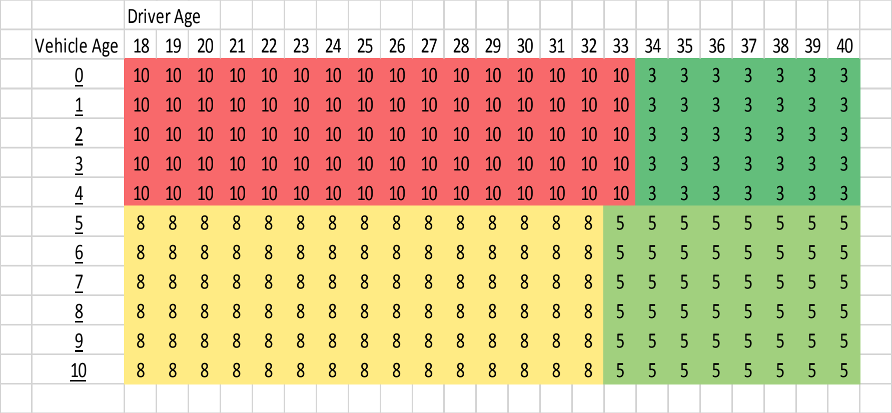
- **Progressive, the insurance company, would like to have a quick way to quote the premium for an insurance policy on a car.**
- **The insurance agent only has 5 minutes to spend on the phone with a potential new customer.**
- **The only information the agent gets is the caller’s age and the caller’s vehicle age.**
- **How can a data scientist help with this problem?**

Data Science Problem

- Suppose the company has data from the insurance policies written in the past. If saved in a data frame, we would have three variables: Premium, Driver Age and Vehicle Age. How can we fit a model so that we can predict the Premium for a future, only knowing the Driver Age and Vehicle Age?

Data Science Problem

- Historical Data



	Driver Age																						
Vehicle Age	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
0	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
1	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
2	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
3	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
4	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
5	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
6	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
7	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
9	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
10	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5

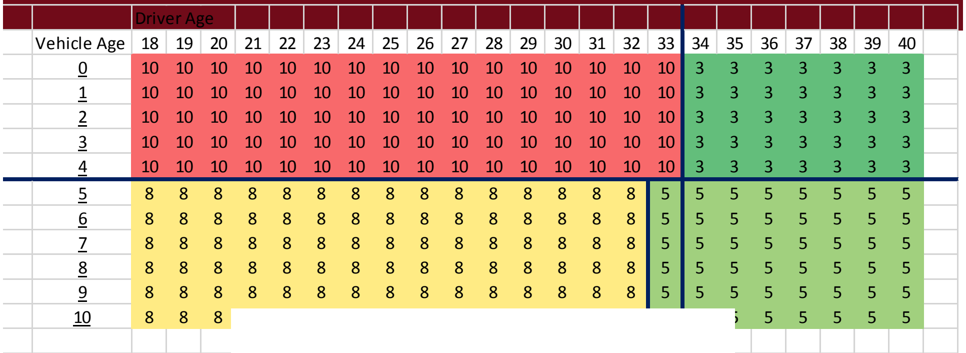
Decision Trees (Predictive task)

- We can build a decision tree. Decision trees are a class of techniques used to characterize the relationship between a response and a collection of covariates. In R, you can fit a decision tree, and then plot it to have a visualization of the tree.

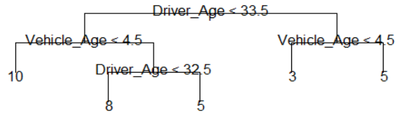
- In R:

- ***library(tree)***
- ***insurance_tree <- tree(Premium ~ Driver_age + Vehicle_age, data = training.cars)***
- ***plot(insurance_tree)***

Decision Trees (Training Set=20% of data)

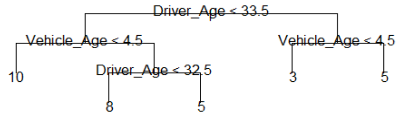


	Driver Age																						
Vehicle Age	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
0	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
1	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
2	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
3	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
4	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	3	3	3	3	3	3	3
5	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
6	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
7	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
9	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	5	5	5	5	5	5	5	5
10	8	8	8															5	5	5	5	5	5



Decision Trees (Testing Set)

```
Predict.tree(insurance_tree, data = testing.cars)
```



In_class3

1) Merge median income dataset and custdata dataset. (Lookup the “merge” R function in help and how to use it).

- Create a new variable called “norm.income” by scaling the “income” variable. Use the appropriate median income for scaling.
- Get numerical summaries of the new variable.
- Comment on a situation when this normalization would make sense (e.g take the new job in that other state or not?).

2) Split the new dataset into a 30% training set and a 70% testing set.