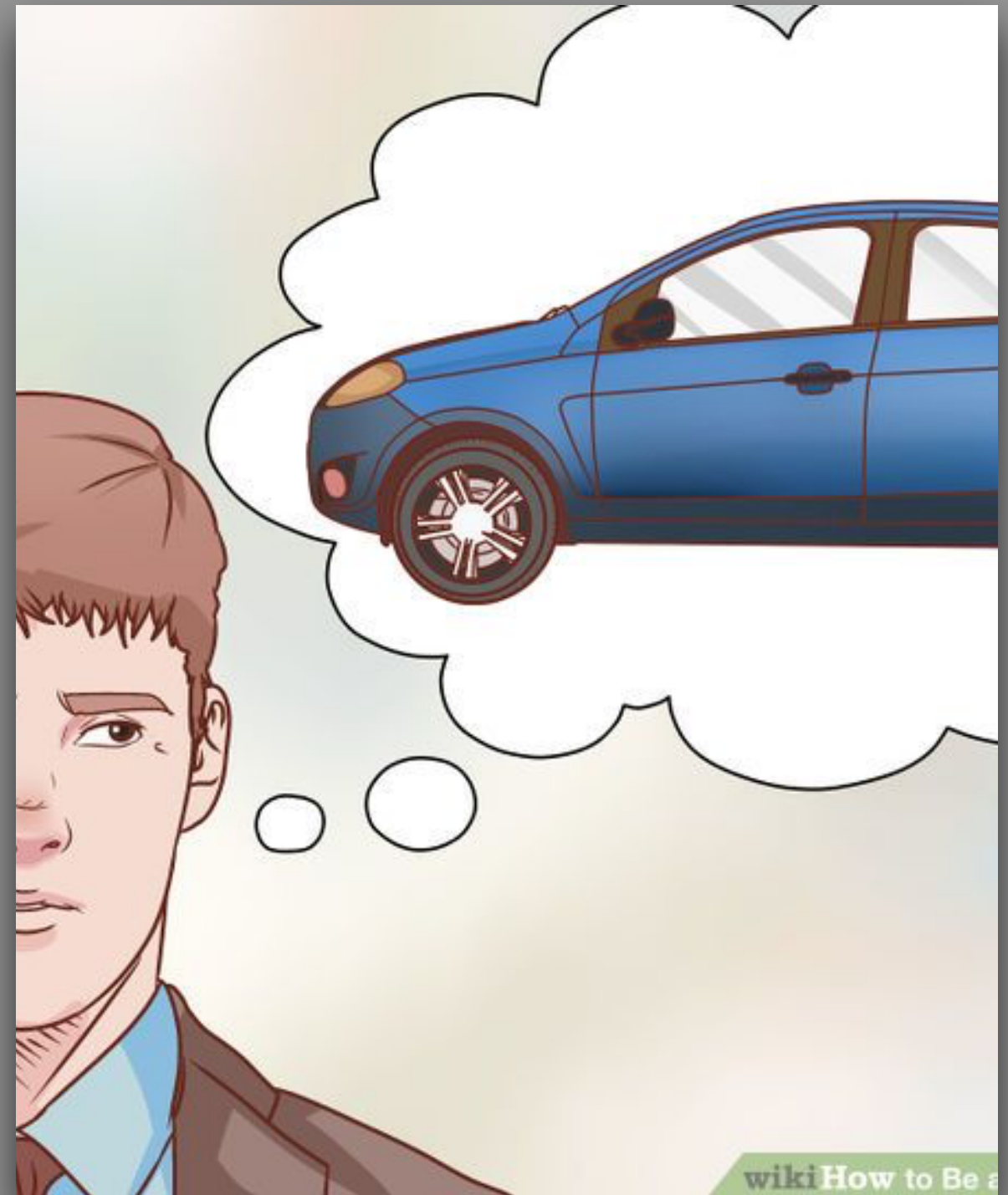


# Predicting Top Customers Using a Classification Tree

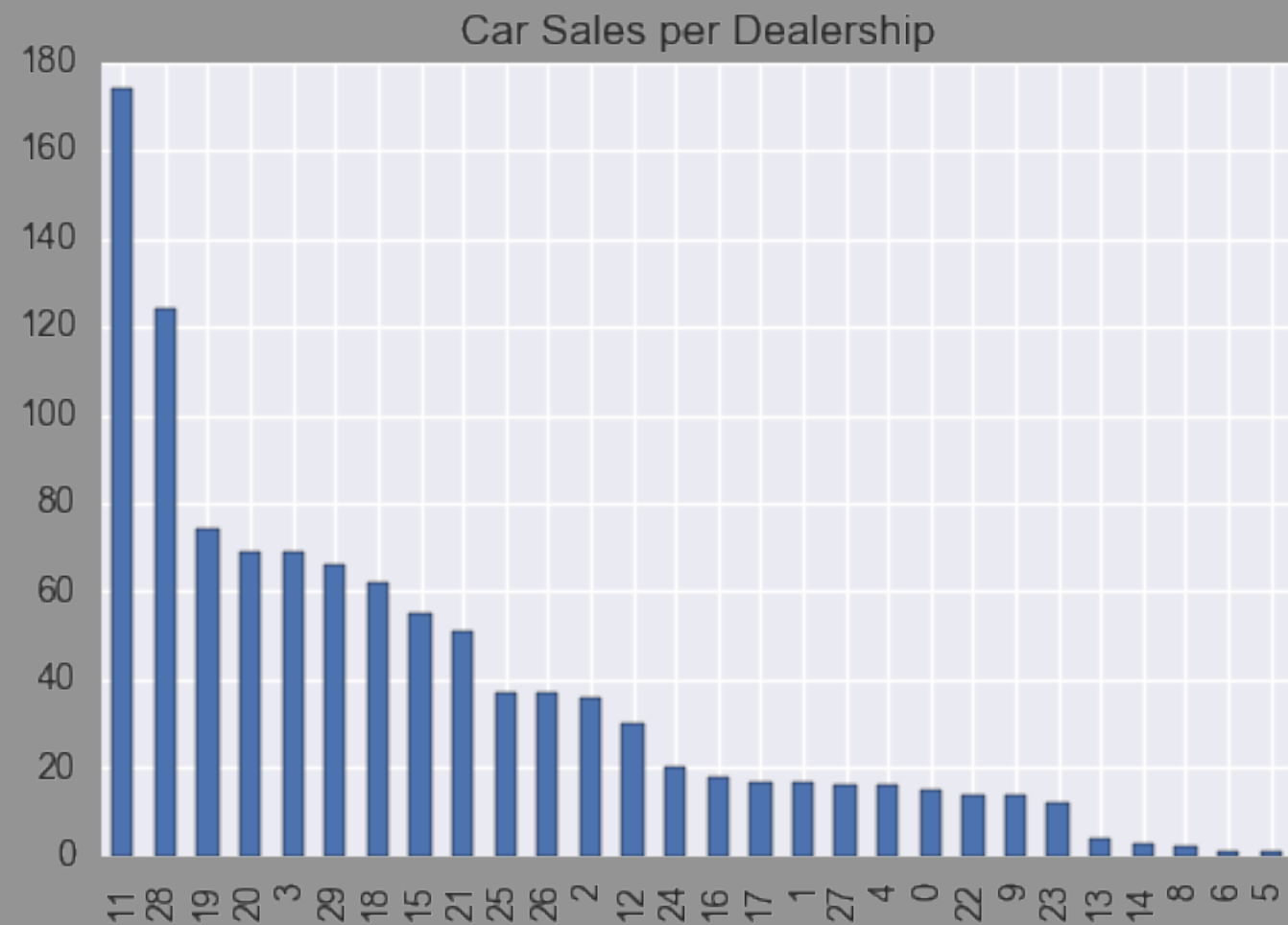
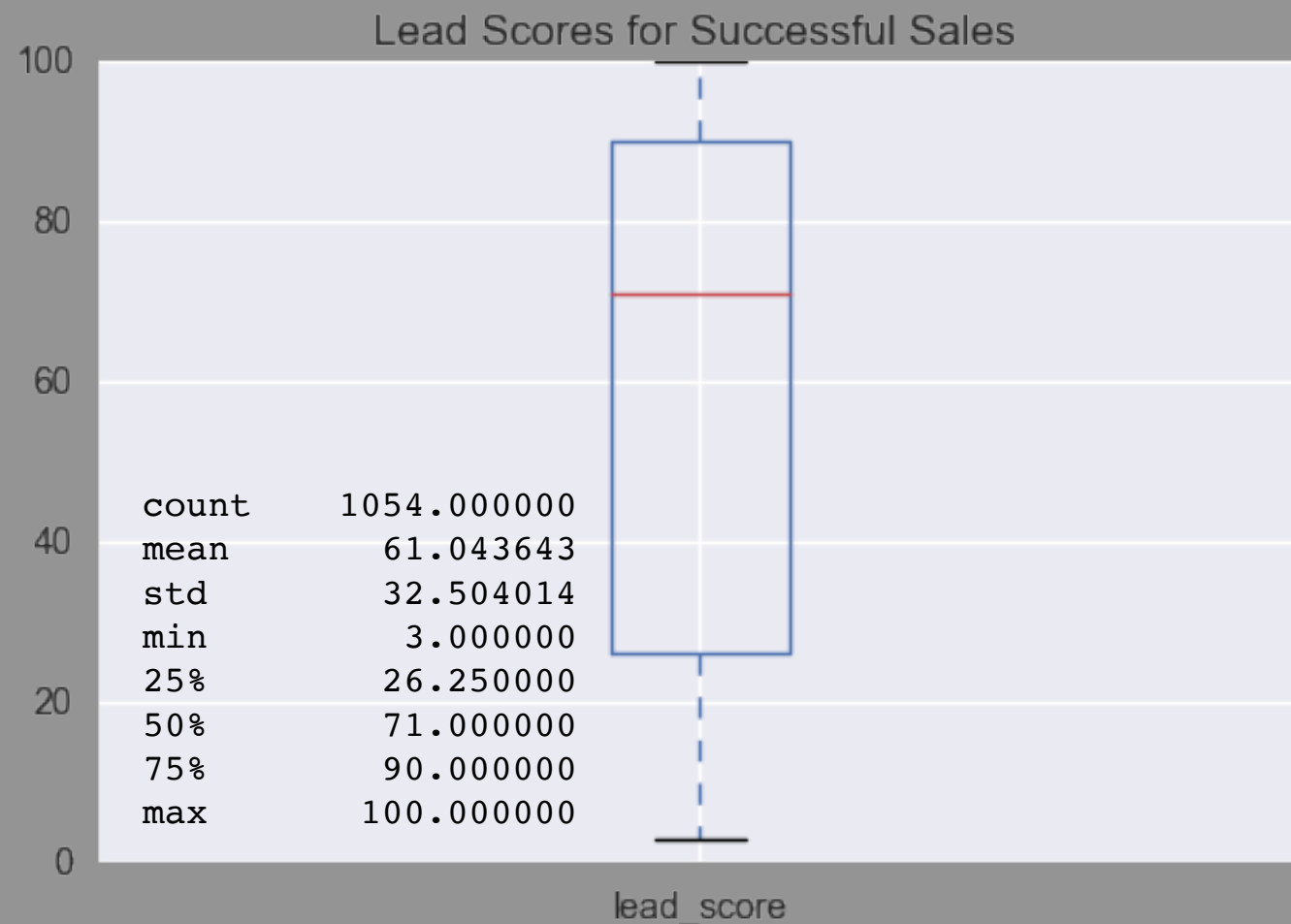
Author: Lauren Washington  
General Assembly SF Fall 2016

# Problem

- Develop a definition for customers who are most likely to purchase a vehicle.
- Predict if a customer will buy a vehicle.
- Develop a definition for a high ranking customer.
- Predict the rank of a customer.
- Develop a function for the probability of obtaining a sale.

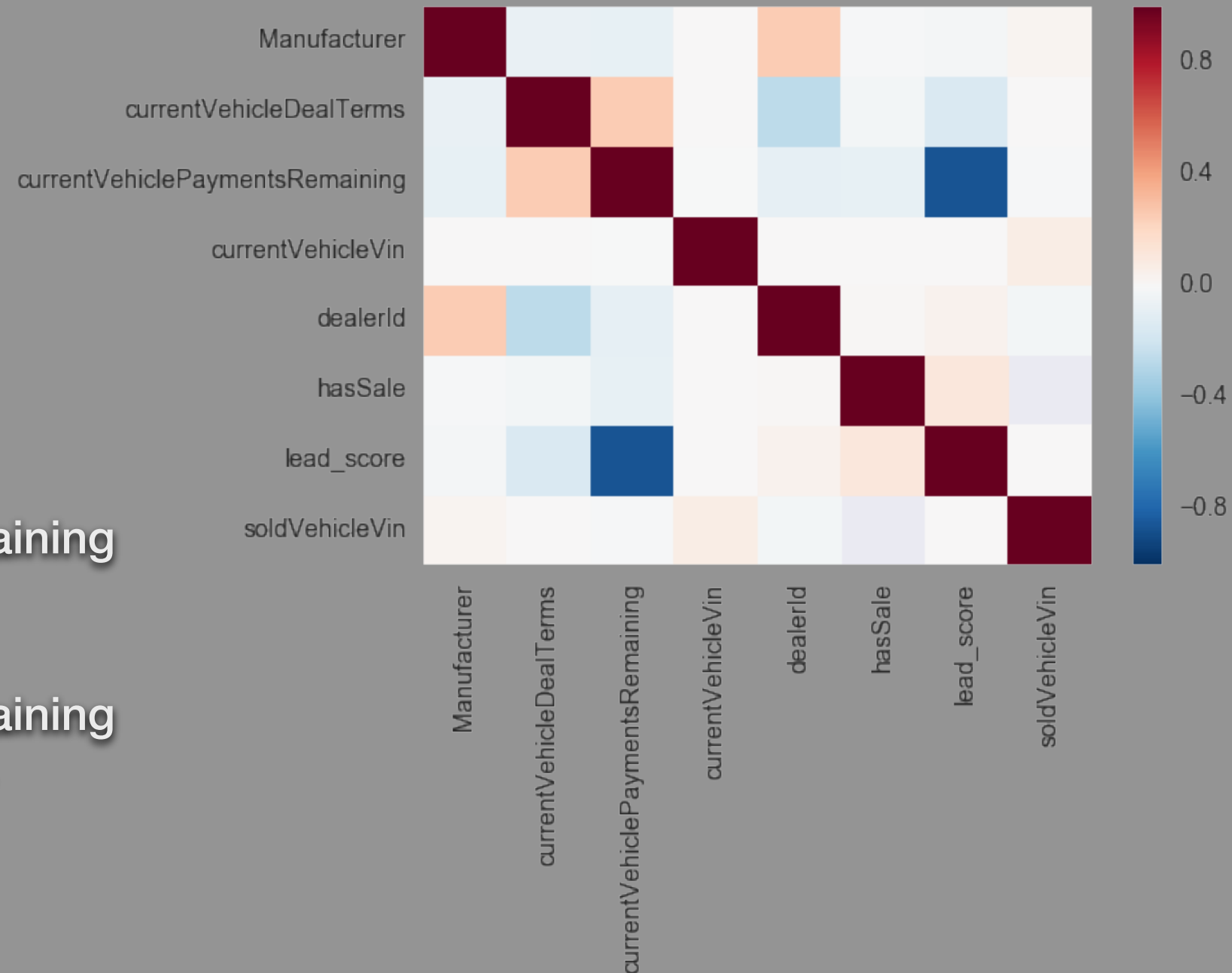


- Mean Lead\_Score is lower than expected and customers w/ a lead\_score of 3 even bought a car.
- It wouldn't be wise to instruct a salesperson to only go after the top lead\_lead scores.
- How can we construct a better prediction for sales?



# Correlations

- Manufacturer and Dealer Id
- currentVehicleDealTerms and DealerId
- currentVehiclePaymentsRemaining and Lead\_Score
- currentVehiclePaymentsRemaining and currentVehicleDealTerms

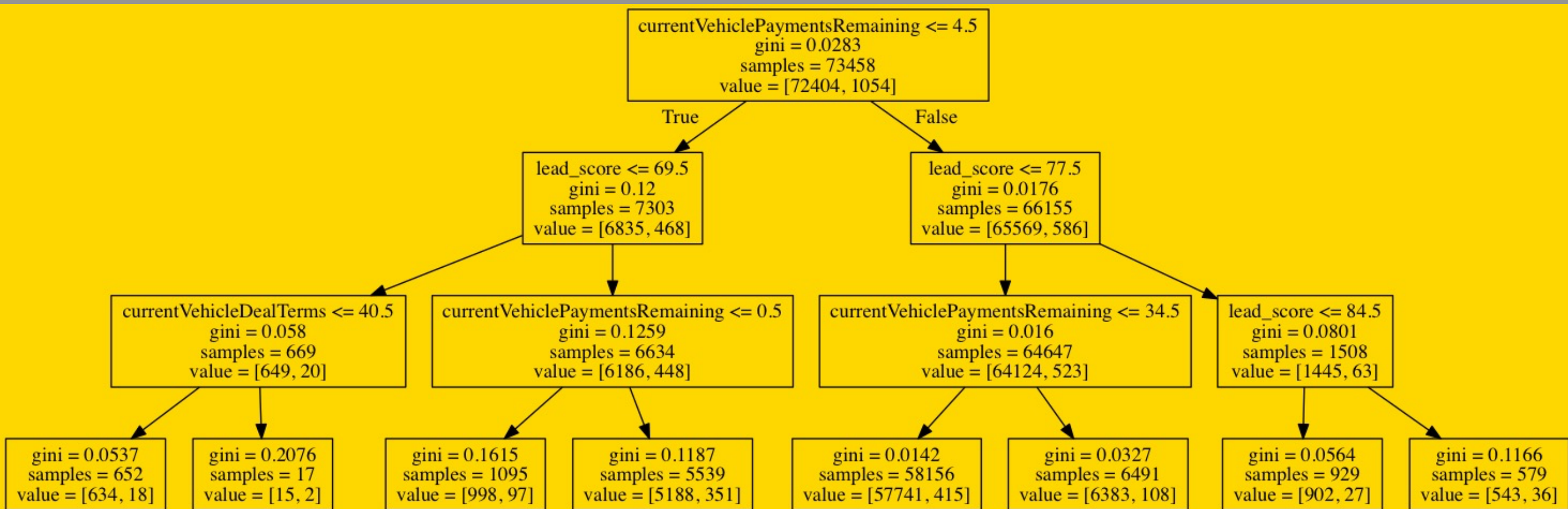


# hasSale Model Fit Scores

MODEL	SCORE
CLASSIFICATION TREE	Null Accuracy/ Max Depth = 3 .98563805
CLASSIFICATION TREE	Max Depth = 2 .98565167

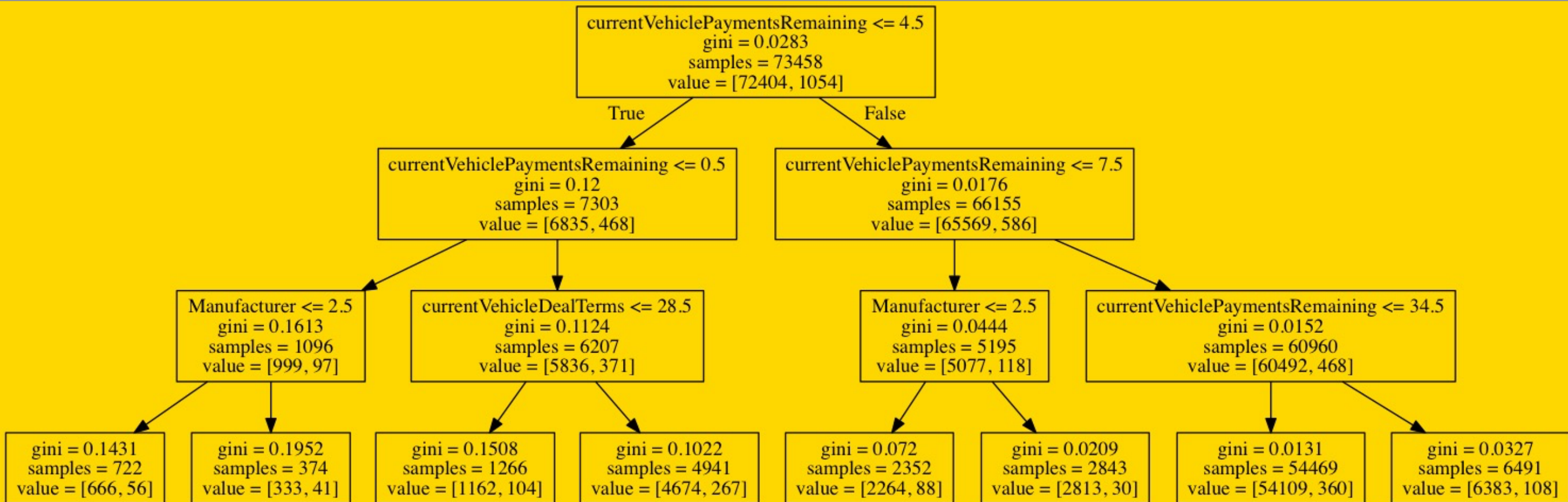


# hasSale Tree Classification (w/ lead\_score)



- if [`currentVehiclePaymentsRemaining <= 4.5`] & [`lead_score <= 69.5`] & [`currentVehicleDealTerms <= 40.5`]
- if [`currentVehiclePaymentsRemaining >= 4.5`] & [`lead_score <= 77.5`] & [`currentVehiclePaymentsRemaining <= 34.5`]

# hasSale Tree Classification (w/o lead\_score)



- if [currentVehiclePaymentsRemaining <= 4.5] & [currentVehiclePaymentsRemaining <= 0.5] & [Manufacturer <= 2.5]
- if [currentVehiclePaymentsRemaining >= 4.5] & [currentVehiclePaymentsRemaining <= 7.5] & [Manufacturer <= 2.5]

# Logistic Regression (Lead Matching)

	FEATURES	COEFICIENT
0	Manufacturer	-0.220447271306
1	currentVehicleDealTerms	-0.0150766560171
2	currentVehiclePaymentsRemaining	-0.0698020516319

- Once the customer segment is identified, the sales team can use a logistic regression function to send the customers less likely to buy to the more seasoned salesman.



# Sales Associate Matching

```
import numpy as np

def get_customer_info():
    if response == 'yes':
        intercept = -2.09049696
        X1 = int(raw_input('What\'s the manufacturer of their current car? '))
        X2 = int(raw_input('What are the current vehicle deal terms? '))
        X3 = int(raw_input('How many payments are remaining on the current vehicle deal terms? '))
        logodds = intercept + X1*(-0.220447271306) + X2*(-0.0150766560171) + X3*(-0.0698020516319)
        odds = np.exp(logodds)
        prob = odds/(1 + odds)
        prob
        if prob < .25:
            print prob
            print ('Give customer to seasoned sales associate.')
        else:
            if prob > .25 & prob < .75:
                print calc
                print('Give cstomer to mid level sales associate.')
            else:
                if prob > .75:
                    print calc
                    print('Let a newbie test their skills :)!')
        else:
            if response == 'no':
                print('Please return when you\'d like to predict a customer sale.')
                print('Enjoy your day!')

print ('Welcome to the Sales predictor!')
a = raw_input('Are you looking to predict a customer outcome? ')
```

```
Documents — -bash — 80x24

Last login: Thu Dec 8 16:13:37 on ttys004
[Laurens-MacBook-Pro:~ laurenwashington$ cd Documents/
[Laurens-MacBook-Pro:Documents laurenwashington$ python SalesPredictor.py
Welcome to the Sales predictor!
Are you looking to predict a customer outcome? yes
What's the manufacturer of their current car? 3
What are the current vehicle deal terms? 27
How many payments are remaining on the current vehicle deal terms? 3
0.0333005205731
Give customer to seasoned sales associate.
Would you like to predict another customer outcome? no
Thank you for using Sales Predictor 1.0 beta!
Laurens-MacBook-Pro:Documents laurenwashington$
```