

Assignment

Contents

Instructions	1
Visualizing Data	2
Modeling Data	3
Cleaning Data	6
Appendix	8

Instructions

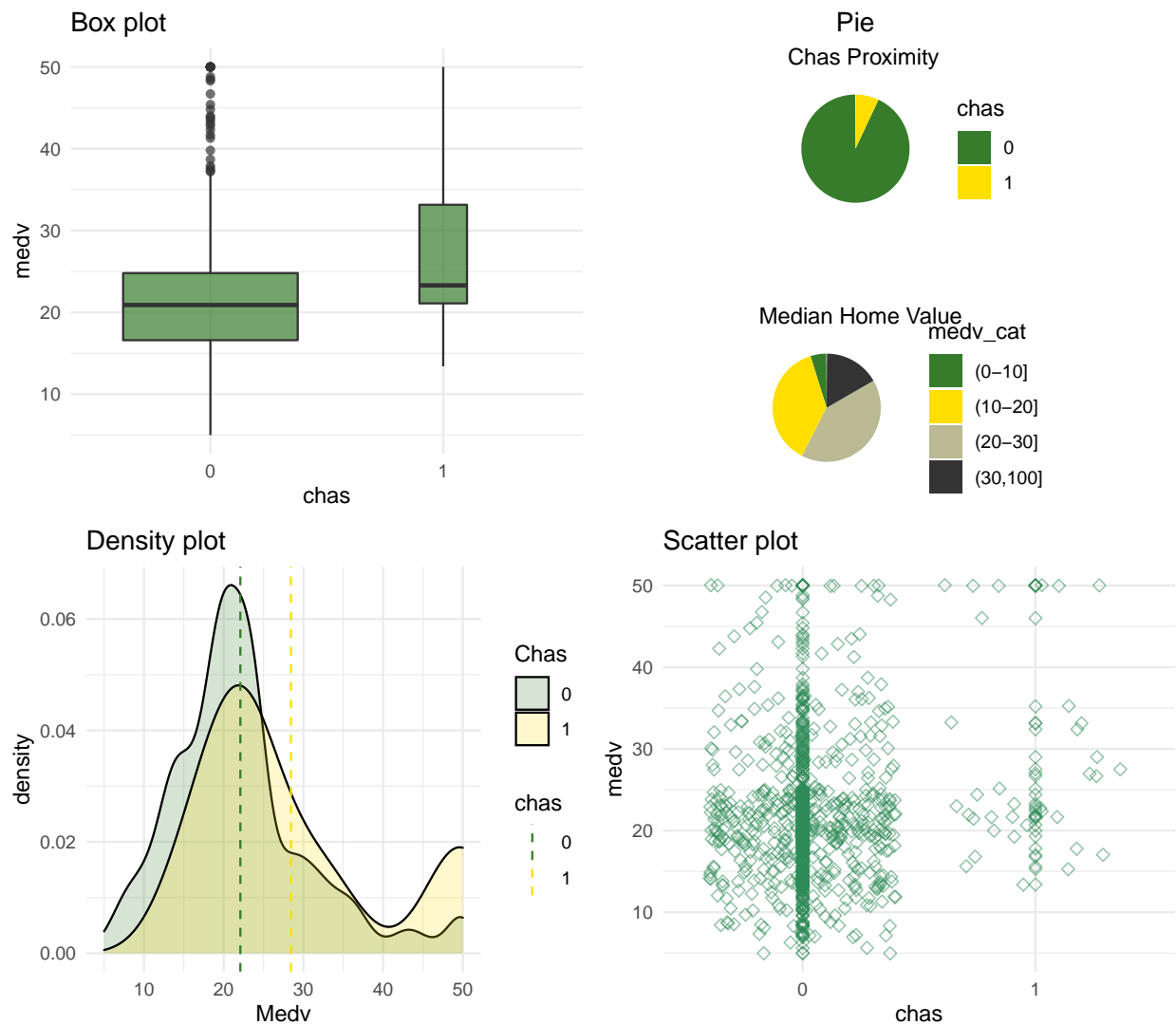
This PDF contains a long form version of the questions, as well as a data appendix which is required to answer a few of the questions. No outside data is needed to answer these questions.

Visualizing Data

This section assesses the ability to interpret and communicate insights. The Boston Housing data in the appendix was used to make the visuals in this section.

Question 1: You are asked to create reports containing visualizations of the Boston Housing data. What tools, software, packages, and/or libraries would you use to create an interactive report? What if the document is required to be printed?

Question 2: Business leaders who will review this report want to understand median home values, Charles River proximity, and the relationship between the two. The following four visuals are considered. Which do you recommend to include in the report and why? (Remember to refer to the appendix for data details.)



Question 3: What changes would you make to the visual you chose in the previous question to make it more interpretable and visually appealing? In your report to business leaders, how would you describe the plot in one sentence?

Modeling Data

This section assesses the ability to think critically about variables and how they can be used to predict a desired outcome. Core competencies include understanding distributions in data, making appropriate data transformations, and selecting an appropriate model. The Boston Housing Data Exploration in the Appendix should be used for this section.

Question 4: *Predicting River Proximity: Using the Boston Housing data, you want to predict which tracts are adjacent to the Charles river (as denoted by the `chas` variable). Based on the data summary in the appendix, what data cleaning, transformations, and/or feature engineering can be used to prepare the data prior to model training?*

Question 5: *Predicting River Proximity: Using the Boston Housing data, you want to predict which tracts are adjacent to the Charles river (as denoted by the `chas` variable). Propose a model to investigate the relationship between the covariates and the `chas` variable. Explain how you would use the model and its output to provide evidence of the strength and confidence in the relationship.*

Question 6: *The realtor thinks that a tract on the Charles River increases median home values by \$5,000. To test the realtor's hypothesis, you created a linear regression model with `chas` as a covariate (Note that `medv` is in \$1,000's of dollars). The coefficient associated with `chas` was 2.87 with a standard error of 0.86. Is there evidence to reject the realtor's claim? Additionally, how would you defend this analysis with the realtor (who has no knowledge of linear models)?*

Question 7: *Predicting Home Value: A local realtor wants to use the data to accurately provide house pricing estimates for clients. Below is the code a data scientist used to fit a random forest model using the Boston Housing data. Critique the approach, point out any errors, and make recommendations for how it can be improved.*

```
# This is R language

data = BostonHousing

# Normalize Variables to improve model performance
vars_to_normalize = c('crim', 'zn', 'indus', 'nox', 'rm',
                      'rad', 'age', 'dis', 'medv')
normalize = function(x){
  return (x - mean(x, na.rm=TRUE))/sd(x, na.rm=TRUE)
}

for (var in vars_to_normalize){
  data[[var]] = normalize(data[[var]])
}

set.seed(1337)

# runif produces a random number between 0 and 1
data["in_test"] = runif(nrow(data)) > .7

get_error = function(model, data){
  data["prediction"] = predict(model, newdata=data)
  return(mean(abs(data[["prediction"]] - data[["medv"]]))))
}

# medv ~. is R shorthand for use medv as the response and all variables as covariates
rf1 = randomForest(medv ~., data = data, ntree = 50)
rf2 = randomForest(medv ~., data = data, ntree = 100)
```

```
rf3 = randomForest(medv ~., data = data, ntree = 250)
rf4 = randomForest(medv ~., data = data, ntree = 500)
rf5 = randomForest(medv ~., data = data, ntree = 1000)

get_error(rf1, data)
> 0.9724985
get_error(rf2, data)
> 0.9945239
get_error(rf3, data)
> 0.9599333
get_error(rf4, data)
> 0.9648902
get_error(rf5, data)
> 0.9622595

rf31 = randomForest(medv ~., data = data, ntree = 250, mtry=3)
rf32 = randomForest(medv ~., data = data, ntree = 250, mtry=4)
rf33 = randomForest(medv ~., data = data, ntree = 250, mtry=5)
rf34 = randomForest(medv ~., data = data, ntree = 250, mtry=6)
rf35 = randomForest(medv ~., data = data, ntree = 250, mtry=7)

get_error(rf31, data)
> 1.047673
get_error(rf32, data)
> 0.9707686
get_error(rf33, data)
> 0.9233799
get_error(rf34, data)
> 0.9104278
get_error(rf35, data)
> 0.9072668

rf351 = randomForest(medv ~., data = data, ntree = 250, mtry=7, nodesize=3)
rf352 = randomForest(medv ~., data = data, ntree = 250, mtry=7, nodesize=5)
rf353 = randomForest(medv ~., data = data, ntree = 250, mtry=7, nodesize=7)
rf354 = randomForest(medv ~., data = data, ntree = 250, mtry=7, nodesize=9)
rf355 = randomForest(medv ~., data = data, ntree = 250, mtry=7, nodesize=11)

get_error(rf351, data)
> 0.822931
get_error(rf352, data)
> 0.8958512
get_error(rf353, data)
> 1.003511
get_error(rf354, data)
> 1.085806
get_error(rf355, data)
> 1.14492

# Final Error Metric:
get_error(rf351, data[data[["in_test"]],])
```

```
> 0.7603227
```

Question 8: What model performance metric was used in the model above to evaluate performance (Question 7)? Propose another function/way of measuring performance and discuss the benefits/drawbacks of each.

Question 9: If you could augment the Boston Housing data with external datasets, what additional data would you want to include, and how would you obtain and integrate it?

Cleaning Data

This section assesses the ability to understand and improve upon existing code, with an emphasis on communication and efficiency. Effective code should accomplish a desired purpose in a clear and effective way that is reproducible between coworkers.

You are working with a colleague to clean a series of datasets. The datasets are a series of csv files with different names. A preview of data1 is shown below along with the code to process the data. All datasets have the same column names, the same datatypes, and contain similar data. Assume that files other than the data are not present in the “unclean” folder.

user	identifier
1	012301A
2	012301A
3	030.1B
4	011323CA
5	012301A

```
# This is python language
data1 = pd.read_csv("C:/Users/Mark/Documents/Project/unclean/data1.csv")
data2 = pd.read_csv("C:/Users/Mark/Documents/Project/unclean/data2.csv")
data3 = pd.read_csv("C:/Users/Mark/Documents/Project/unclean/data3.csv")
data4 = pd.read_csv("C:/Users/Mark/Documents/Project/unclean/data4.csv")

# Code to filter from Question 11 would go on these lines
#
#

for df in [data1,data2,data3,data4]:
    df["identifier2"] = df["identifier"]
    # for row in each row of df
    for i,row in df.iterrows():
        current_string = row["identifier2"]
        new_string = ''
        for s in current_string:
            # For each character in current_string, keep alphabetic
            if s not in ["0","1","2","3","4","5","6","7","8","9"]:
                new_string = new_string + s

        df.loc[i,'identifier'] = new_string
        df.drop(columns = 'identifier2',axis = 1,inplace = True)

final_df = pd.concat([data1,data2,data3,data4])
final_df.to_csv("C:/Users/Mark/Documents/Project/final_data/final_data.csv", index = False)
```

Question 10: Please describe the task in a way such that a non technical co-worker can understand the purpose of the code above.

Question 11: You are considering filtering out all rows where “user” is less than 2. Will the code below accomplish the stated task?

```
#This is python language
dataframes = [data1,data2,data3,data4]
for df in dataframes:
    df = df[(df["user"] <2 )]
```

Question 12: *The stakeholder you're working with needs to read in 50 more files that have the same format. How might you improve on the approach above? Think about reproducibility, redundancy, and code efficiency. Assume the filtering code will not be used.*

Question 13: *Propose an alternative set of code (Python) that can improve on the current code.*

```
# Python code block  
# Your Code Here
```

Appendix

Boston Housing Data

The Boston Housing data was collected on census tracts from Boston in the mid 1970's. This section provides an exploratory analysis of the variables in the dataset that can be used to answer questions from the Visualization and Modeling sections of the assessment.

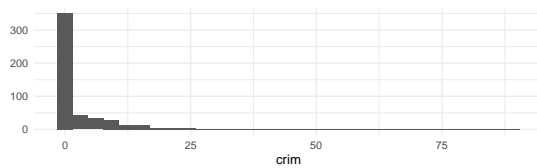
Boston Housing Data Sample

crim	zn	indus	chas	nox	rm	age	dis	rad	medv
0.00632	18	2.31	0	0.538	6.575	65.2	4.0900	1	24.0
0.02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	21.6
0.02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	34.7
0.03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	33.4
0.06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	36.2

Boston Housing Variable Descriptions

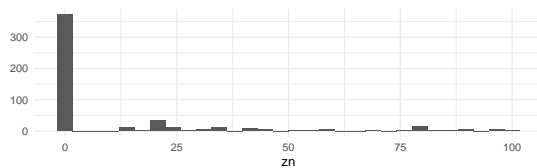
crim - Per Capita crime rate by tract

Below, find the distribution of **crim**. There are no other covariates to **crim** with a pearson correlation above 0.5 or below -0.5.



zn - Percentage of residential land zoned for lots > 25,000 sq.ft.

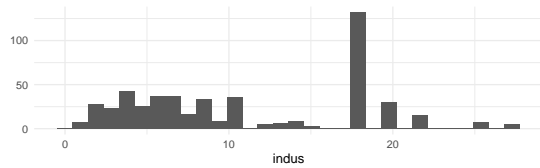
Below, find the distribution of **zn**, as well as covariates to **zn** that have a pearson correlation above .5 or below -.5.



	indus	nox	age	dis
Cor.	-0.5338282	-0.5166037	-0.5695373	0.6644082

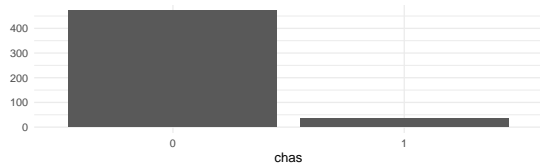
indus - Percentage of non-retail business acres per town

Below, find the distribution of **indus**, as well as covariates to **indus** that have a pearson correlation above .5 or below -.5.



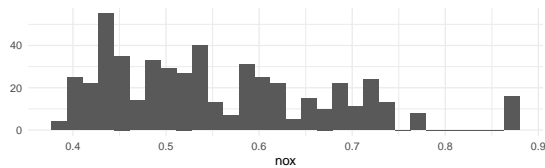
	zn	nox	age	dis	rad
Cor.	-0.5338282	0.7636514	0.6447785	-0.708027	0.5951293

chas - Charles River dummy variable (1 - tract bounds the river, 0 - tract does not bind river)



nox - Nitric Oxide concentration (in ppm)

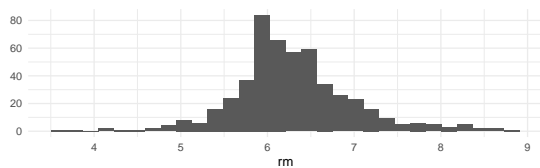
Below, find the distribution of **nox**, as well as covariates to **nox** that have a pearson correlation above .5 or below -.5.



	zn	indus	age	dis	rad
Cor.	-0.5166037	0.7636514	0.7314701	-0.7692301	0.6114406

rm - Average rooms per dwelling

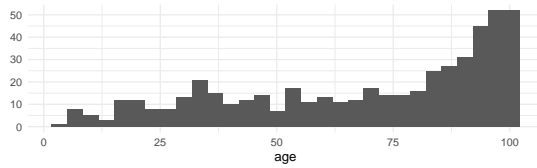
Below, find the distribution of **rm**, as well as covariates to **rm** that have a pearson correlation above .5 or below -.5.



	medv
Cor.	0.6953599

age - Percentage of owner-occupied units built prior to 1940

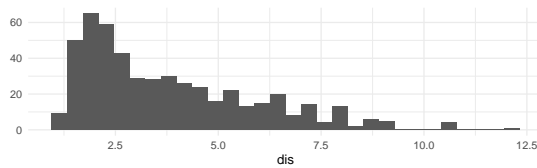
Below, find the distribution of **age**, as well as covariates to **age** that have a pearson correlation above .5 or below -.5.



	zn	indus	nox	dis
Cor.	-0.5695373	0.6447785	0.7314701	-0.7478805

dis - weighted distances to five Boston employment centers

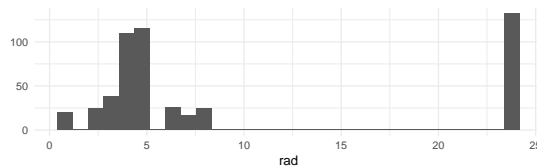
Below, find the distribution of **dis**, as well as covariates to **dis** that have a pearson correlation above .5 or below -.5.



	zn	indus	nox	age
Cor.	0.6644082	-0.708027	-0.7692301	-0.7478805

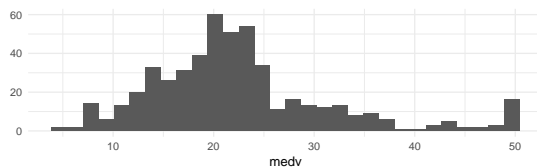
rad - Index of access to radial highways

Below, find the distribution of **rad**. There are no other covariates to **rad** with a pearson correlation above 0.5 or below -0.5



medv - median value of owner-occupied homes in \$1,000's

Below, find the distribution of **medv**, as well as covariates to **medv** that have a pearson correlation above .5 or below -.5.



	rm
Cor.	0.6953599