CS 696 Intro to Big Data: Tools and Methods
Fall Semester, 2017
Doc 14 Visualizations
Oct 24, 2017

Copyright ©, All rights reserved. 2017 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (http://www.opencontent.org/opl.shtml) license defines the copyright on this document.

Agile Software Development

Iterative and incremental software development

Scrum 1995

Crystal Clear 1996

Extreme programming (XP) 1996

Feature-driven development 1997

Manifesto for Agile Software Development

2001

Value

Individuals and Interactions more than processes and tools

Working Software more than comprehensive documentation

Customer Collaboration more than contract negotiation

Responding to Change more than following a plan

Agile software development principles

Customer satisfaction by early and continuous delivery of valuable software

Welcome changing requirements, even in late development

Working software is delivered frequently (weeks rather than months)

Close, daily cooperation between business people and developers

Projects are built around motivated individuals, who should be trusted

Face-to-face conversation is the best form of communication (co-location)

Working software is the primary measure of progress

Sustainable development, able to maintain a constant pace

Continuous attention to technical excellence and good design

Simplicity—the art of maximizing the amount of work not done—is essential

Best architectures, requirements, and designs emerge from self-organizing teams

Regularly, the team reflects on how to become more effective, and adjusts accordingly

A manifesto for Agile data science

https://www.oreilly.com/ideas/a-manifesto-for-agile-data-science

http://tinyurl.com/y8lzavxn

The manifesto focuses on how to think

The key is that you approach data science in an active and dynamic way

A manifesto for Agile data science

Iterate, iterate, iterate

Insight comes from the 25th query in a chain of queries, not the first one

Data tables have to be parsed, formatted, sorted, aggregated, and summarized before they can be understood

A manifesto for Agile data science

Ship intermediate output

Often left at the end of a sprint with things that aren't complete

In Agile data science, we document and share the incomplete assets we create as we work

We commit all work to source control

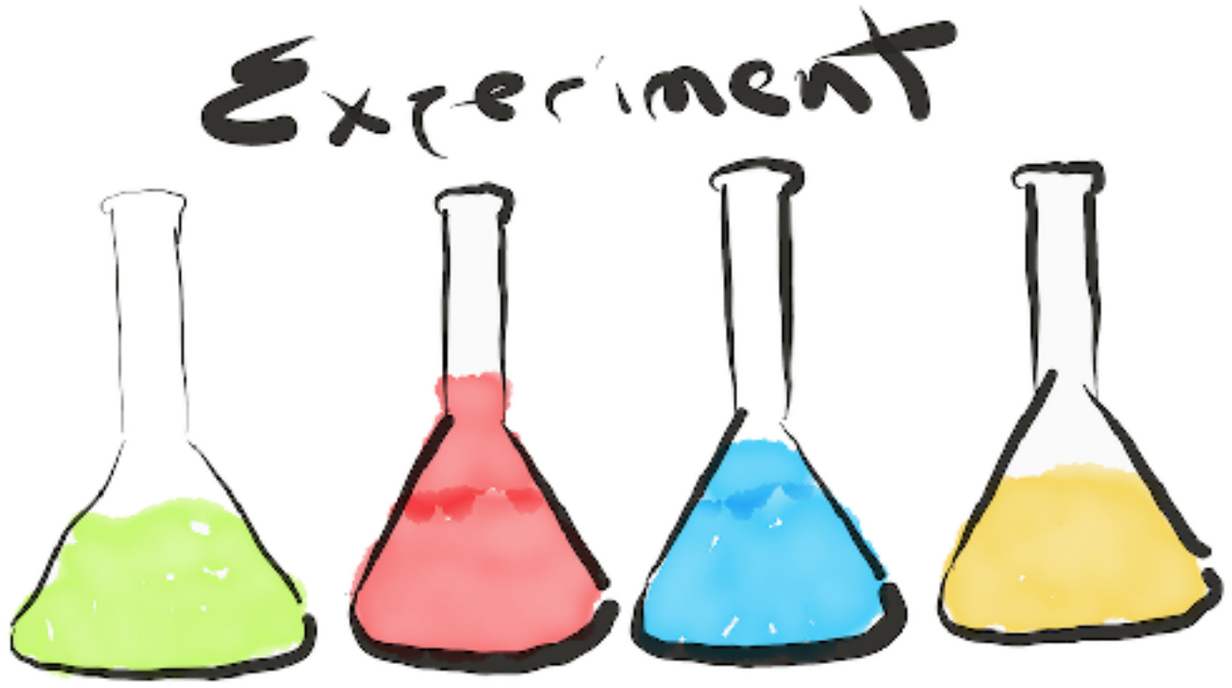
We share this work with teammates and, as soon as possible, with end users

A manifesto for Agile data science

Perform experiments, not tasks

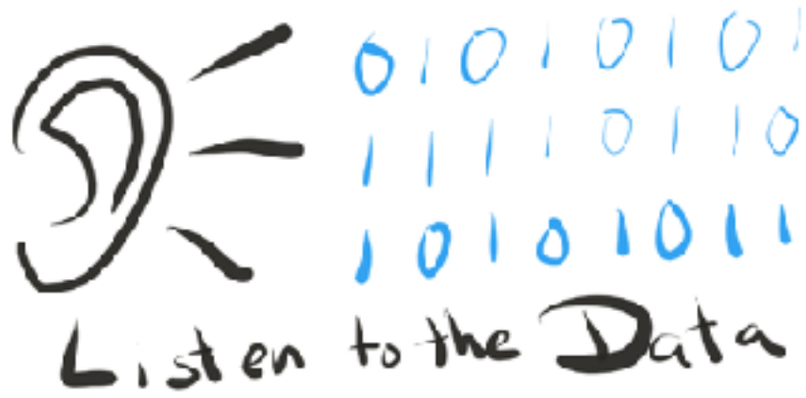
Data science differs from software engineering in that it is part science, part engineering

In any given task, we must iterate to achieve insight, and these iterations can best be summarized as experiments



A manifesto for Agile data science

Listen to the data



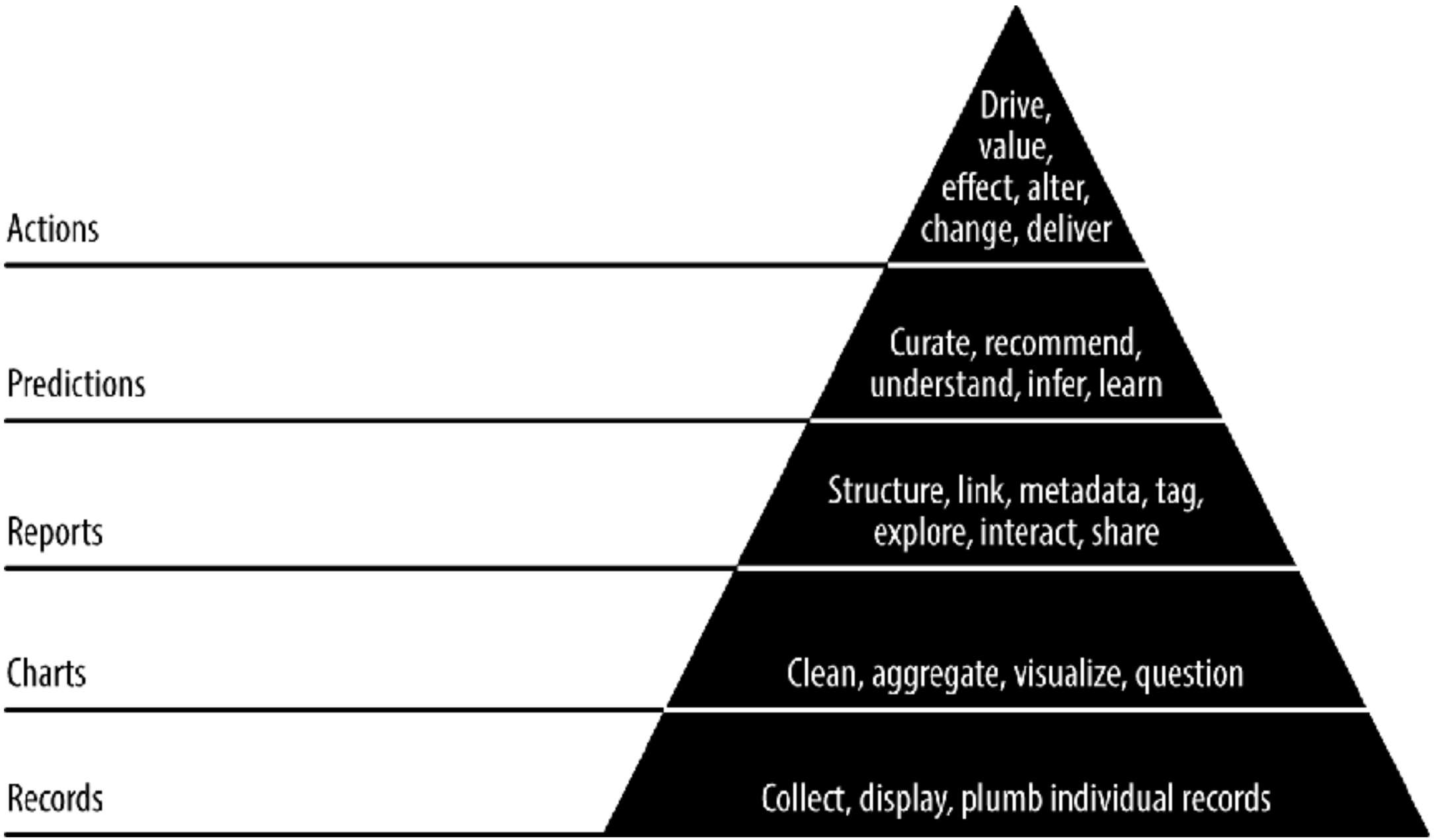
What is possible is as important as what is intended

Without understanding what the data “has to say” about any feature, the product owner can’t do a good job

The data’s opinion must always be included in product discussions, which means that they must be grounded in **visualization** through exploratory data analysis in the internal application that becomes the focus of our efforts

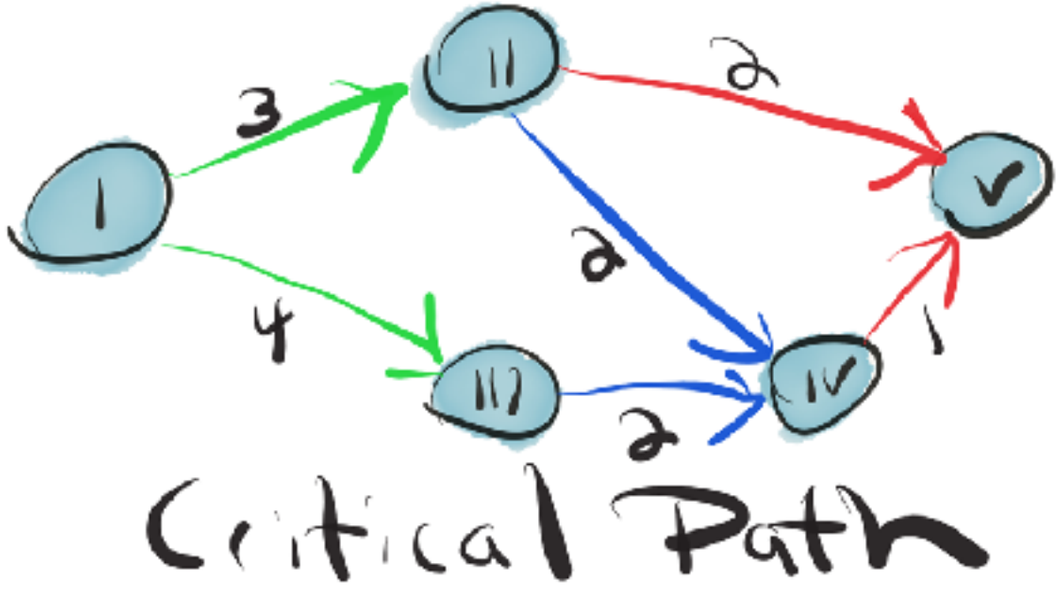
A manifesto for Agile data science

Respect the data-value pyramid



A manifesto for Agile data science

Find the critical path



To maximize our odds of success, we should focus most of our time on that aspect of our application that is most essential to its success. But which aspect is that? This must be discovered through experimentation. Analytics product development is the search for and pursuit of a moving goal.

A manifesto for Agile data science

Get meta

The focus is on documenting the analytics process as opposed to the end state or product we are seeking. This lets us be Agile and ship intermediate content as we iteratively climb the data-value pyramid to pursue the critical path to a killer product.

Spark Notebooks

http://tinyurl.com/ydbqqp46

Apache Zeppelin

PROS	CONS
Good integration with Spark	No tab-completion
Available on AWS Elastic MapReduce (EMR)	

Jupyter-scala

PROS	CONS
Notebooks saved as .ipynb	Has some minor bugs
Tab-completion available	
Easy set up	
No frills, just basic functionality	

Spark Notebooks

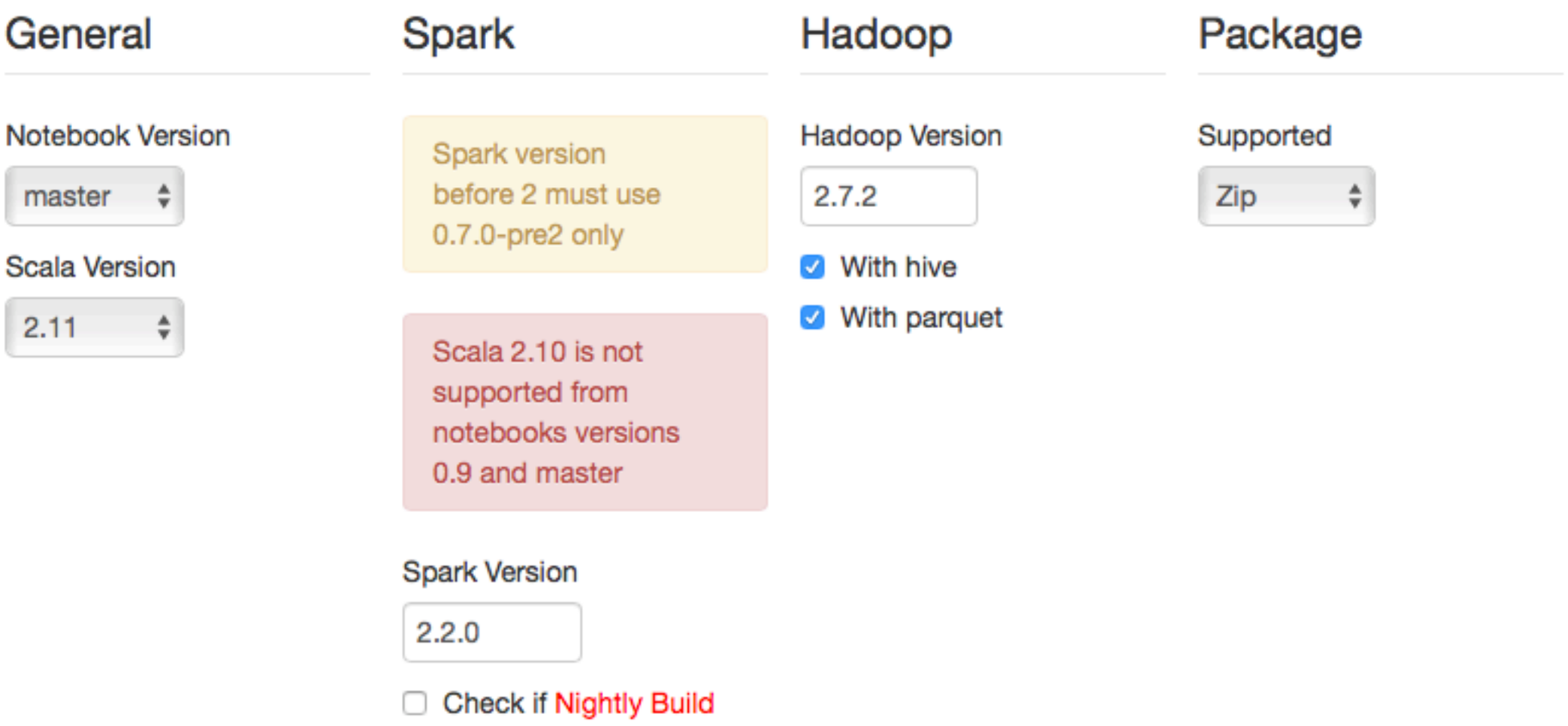
http://tinyurl.com/ydbqqp46

Spark-notebook

PROS	CONS
Very good integration with Spark	A lot of functionality, feels a little bloated
Tab-completion available	
Easy set up: download, extract and run	
Comes with tons of examples	
Lots of charting capabilities	

Spark-notebook

https://github.com/spark-notebook/spark-notebook



Datasets

UC Irvine Machine Learning Repository http://archive.ics.uci.edu/ml/index.php

R Datasets

Amazon Public Data Sets https://aws.amazon.com/public-datasets/

UC Irvine Machine Learning Repository

Browse Through: **394** Data Sets

Table View List View

Default Task
- Classification (289)
- Regression (74)
- Clustering (67)
- Other (54)

Attribute Type
- Categorical (37)
- Numerical (244)
- Mixed (55)

Data Type
- Multivariate (306)
- Univariate (16)
- Sequential (40)
- Time-Series (75)
- Text (37)
- Domain-Theory (22)
- Other (21)

Area
- Life Sciences (89)
- Physical Sciences (47)
- CS / Engineering (129)
- Social Sciences (23)
- Business (25)
- Game (10)
- Other (67)

Name	Data Types	Default Task	Attribute Types	# Instances	# Attributes	Year
Abalone	Multivariate	Classification	Categorical, Integer, Real	4177	8	1995
Adult	Multivariate	Classification	Categorical, Integer	48842	14	1996
Annealing	Multivariate	Classification	Categorical, Integer, Real	798	38	
Anonymous Microsoft Web Data		Recommender-Systems	Categorical	37711	294	1998
Arrhythmia	Multivariate	Classification	Categorical, Integer, Real	452	279	1998
Artificial Characters	Multivariate	Classification	Categorical, Integer, Real	6000	7	1992

R Datasets

Datasets collected to use to learn statistics & use R

Commonly used

List https://vincentarelbundock.github.io/Rdatasets/datasets.html

Package	Item	Title	Rows	Cols	csv	doc
datasets	AirPassengers	Monthly Airline Passenger Numbers 1949-1960	144	2	CSV	DOC
datasets	BJsales	Sales Data with Leading Indicator	150	2	CSV	DOC
datasets	BOD	Biochemical Oxygen Demand	6	2	CSV	DOC
datasets	CO2	Carbon Dioxide Uptake in Grass Plants	237	2	CSV	DOC
datasets	Formaldehyde	Determination of Formaldehyde	6	2	CSV	DOC
datasets	HairEyeColor	Hair and Eye Color of Statistics Students	4	4	CSV	DOC
datasets	InsectSprays	Effectiveness of Insect Sprays	72	2	CSV	DOC
datasets	JohnsonJohnson	Quarterly Earnings per Johnson & Johnson Share	84	2	CSV	DOC
datasets	LakeHuron	Level of Lake Huron 1875-1972	98	2	CSV	DOC
datasets	LifeCycleSavings	Intercountry Life-Cycle Savings Data	50	5	CSV	DOC
datasets	Nile	Flow of the River Nile	100	2	CSV	DOC
datasets	OrchardSprays	Potency of Orchard Sprays	64	4	CSV	DOC
datasets	PlantGrowth	Results from an Experiment on Plant Growth	30	2	CSV	DOC
datasets	Puromycin	Reaction Velocity of an Enzymatic Reaction	23	3	CSV	DOC
datasets	Titanic	Survival of passengers on the Titanic	1313	6	CSV	DOC
datasets	ToothGrowth	The Effect of Vitamin C on Tooth Growth in Guinea Pigs	60	3	CSV	DOC
datasets	UCBAdmissions	Student Admissions at UC Berkeley	2	2	CSV	DOC
datasets	UKDriverDeaths	Road Casualties in Great Britain 1969-84	192	2	CSV	DOC
datasets	UKgas	UK Quarterly Gas Consumption	108	2	CSV	DOC
datasets	USAccDeaths	Accidental Deaths in the US 1973-1978	72	2	CSV	DOC

Dwell Times on Web sites

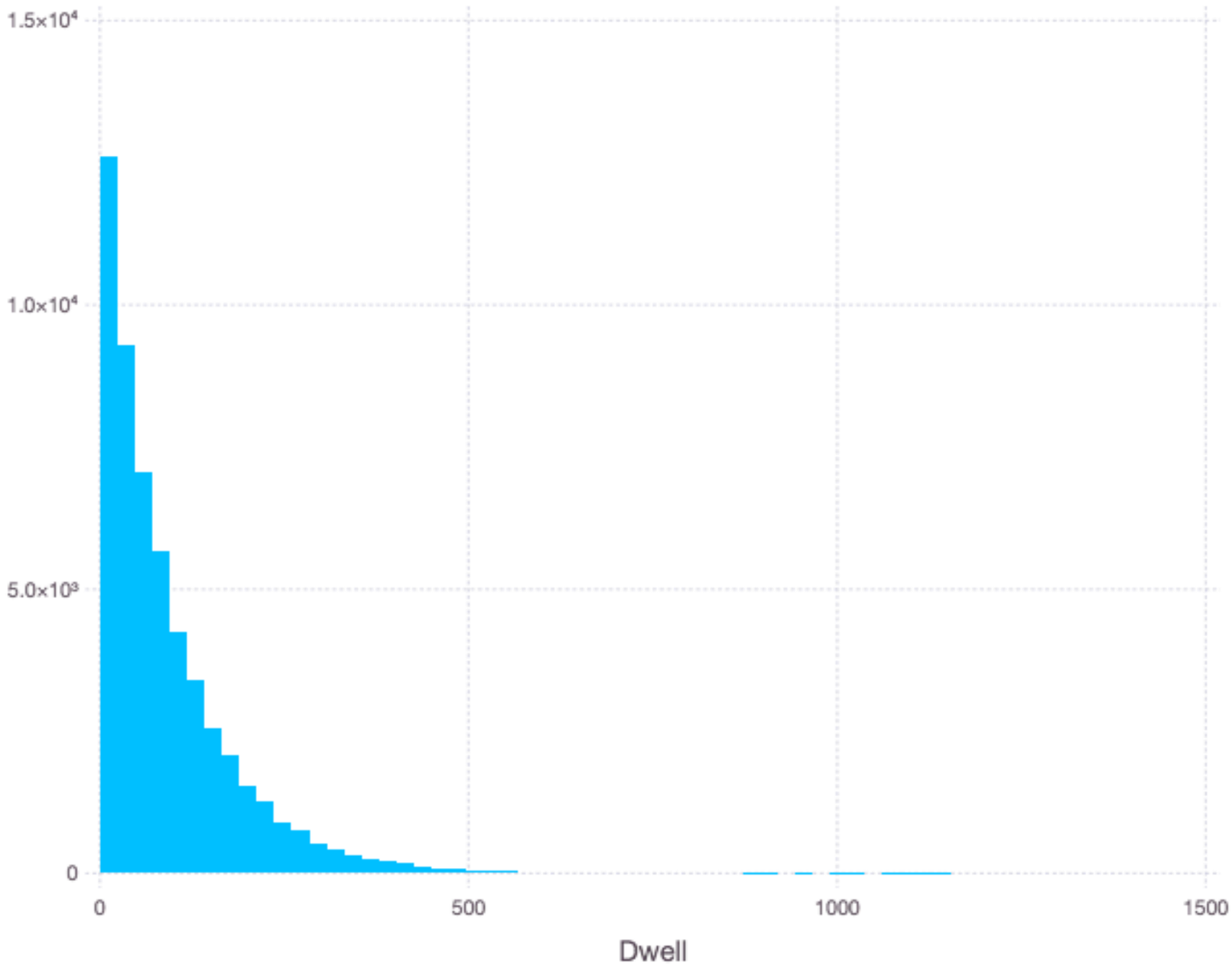
Look at Dwell data of website

Don’t know the distribution of the dwell times

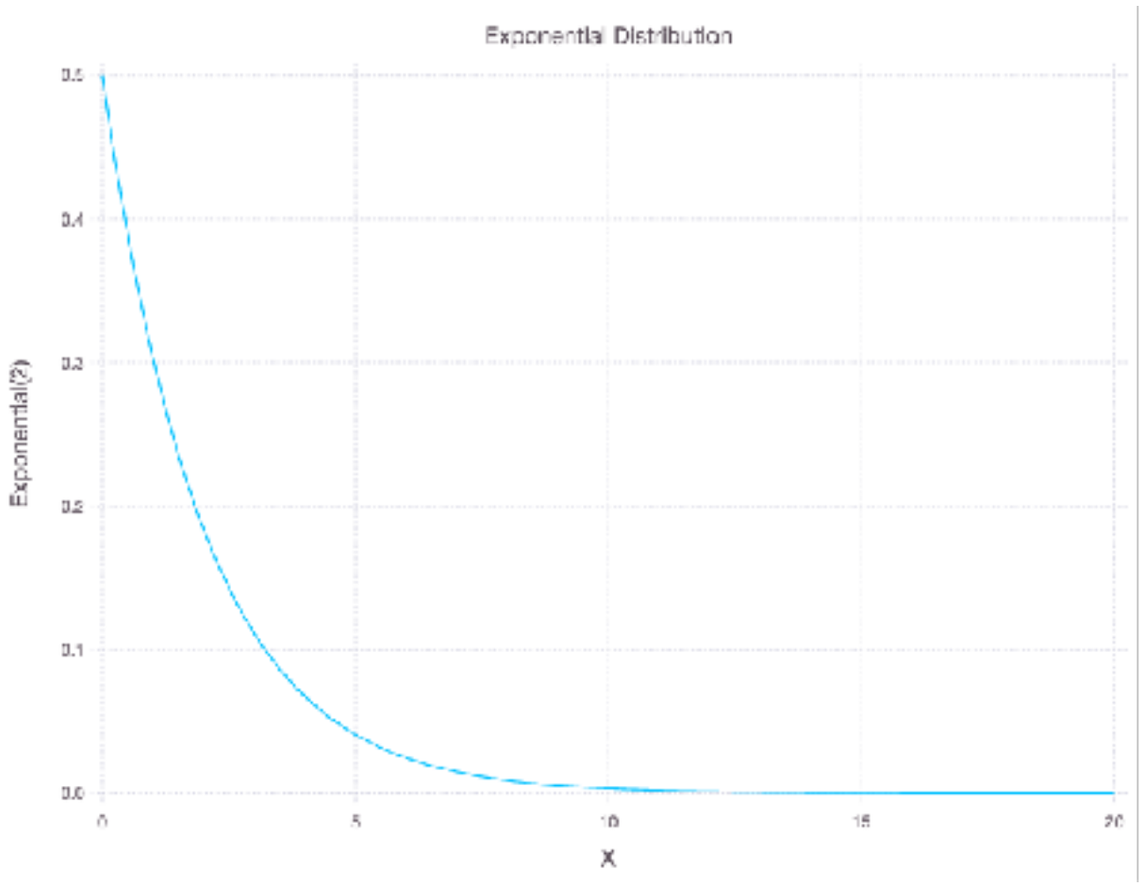
But daily mean of dwell times will be normally distributed

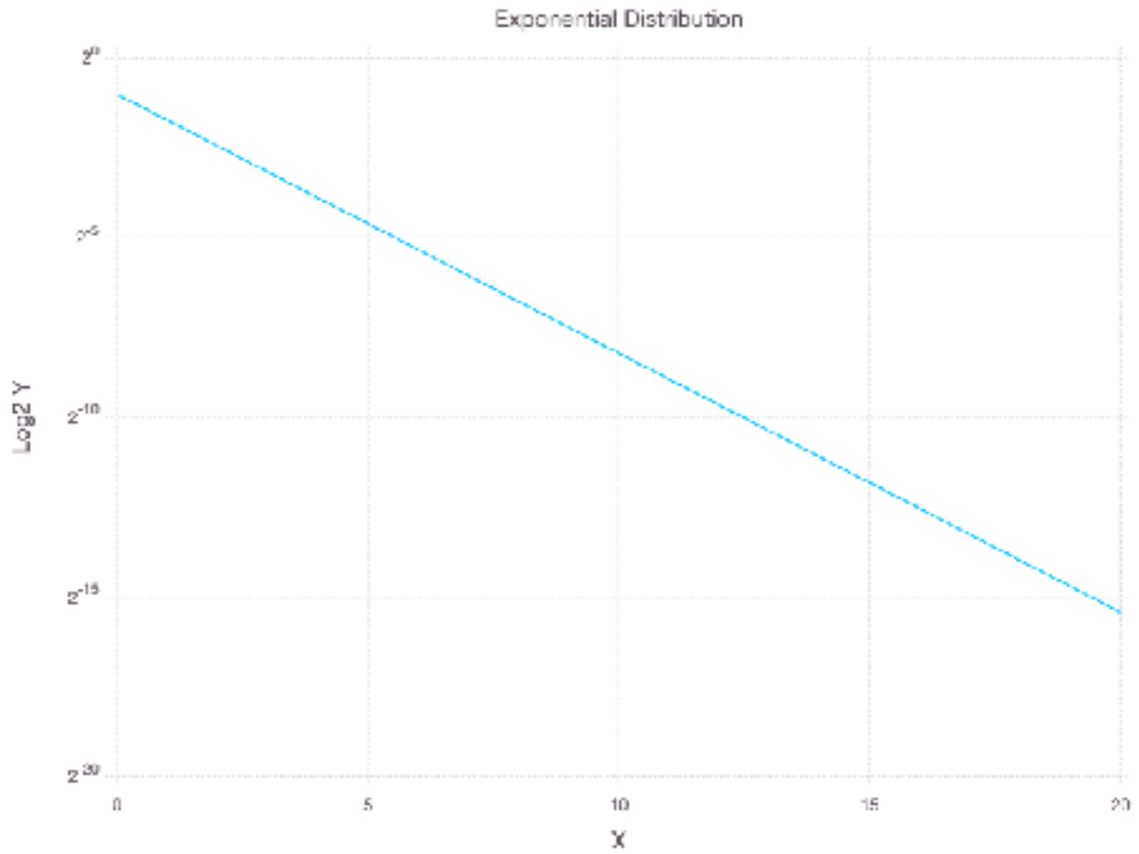
Dwell Times

plot(dwell_times, x="Dwell", Geom.histogram(bincount = 50))



Exponential Distribution

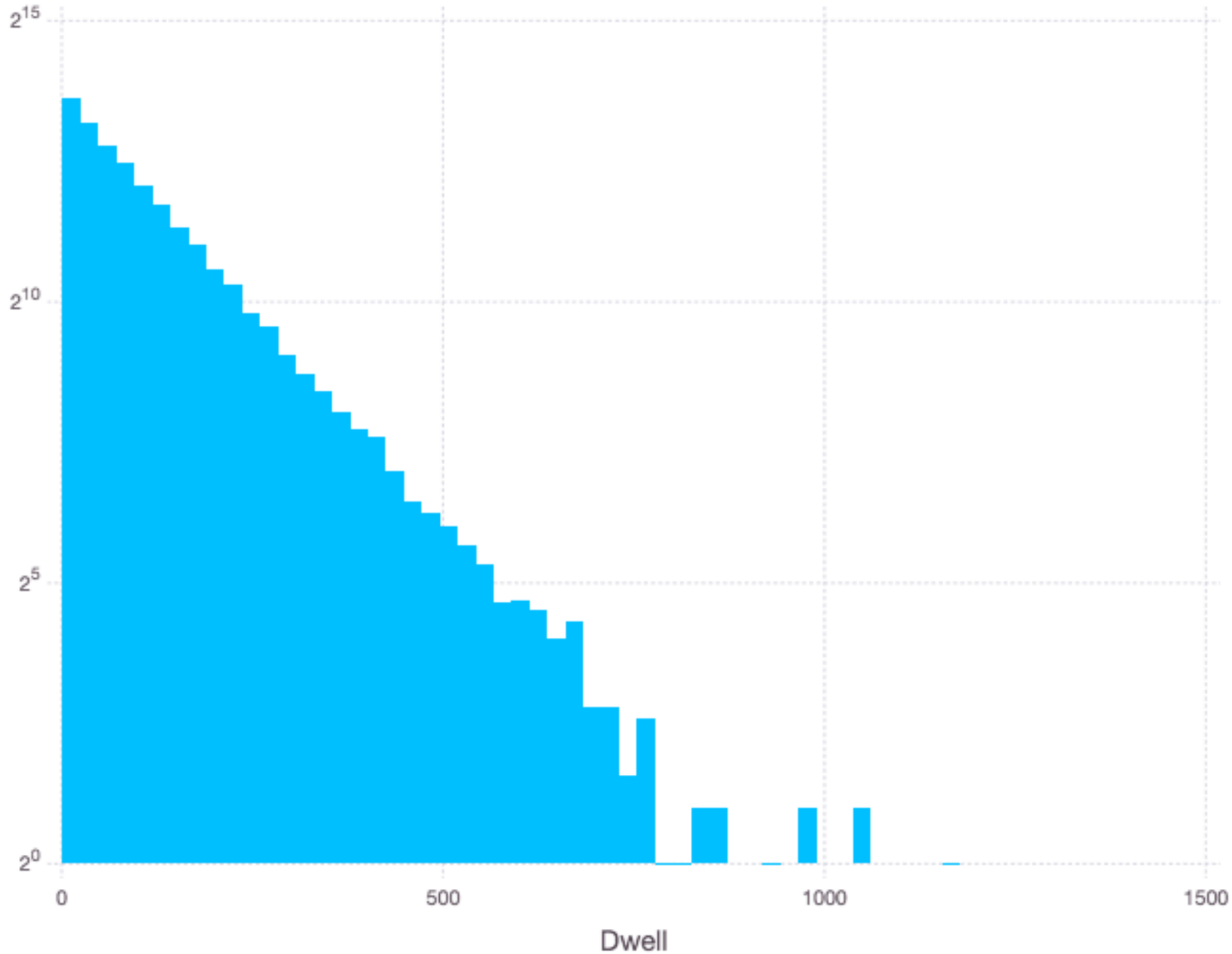




Log2(Y)

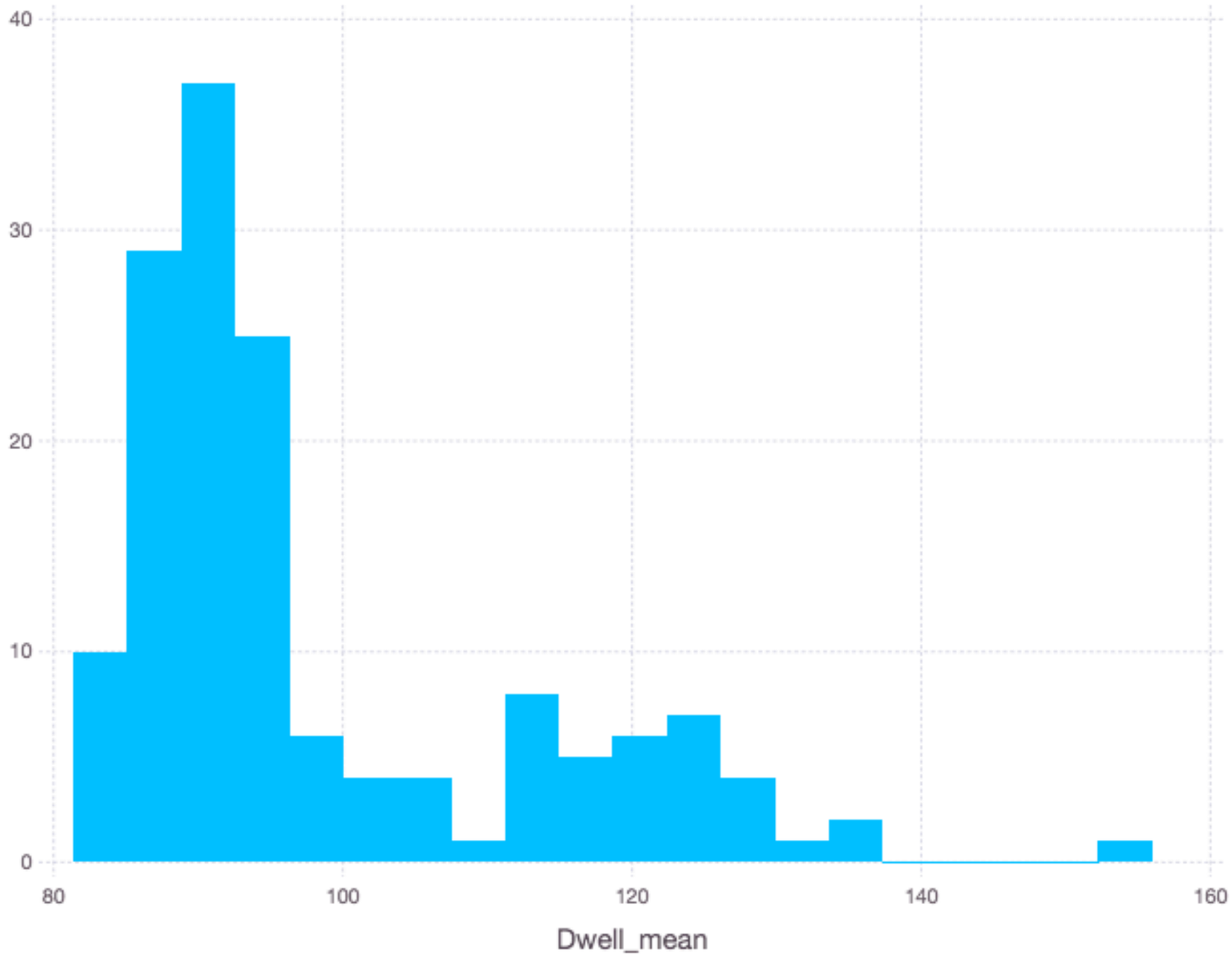
Log Scale - So Dwell Time is Exponential Dist.

plot(dwell_times, x="Dwell", Geom.histogram(bincount = 50), Scale.y_log2)

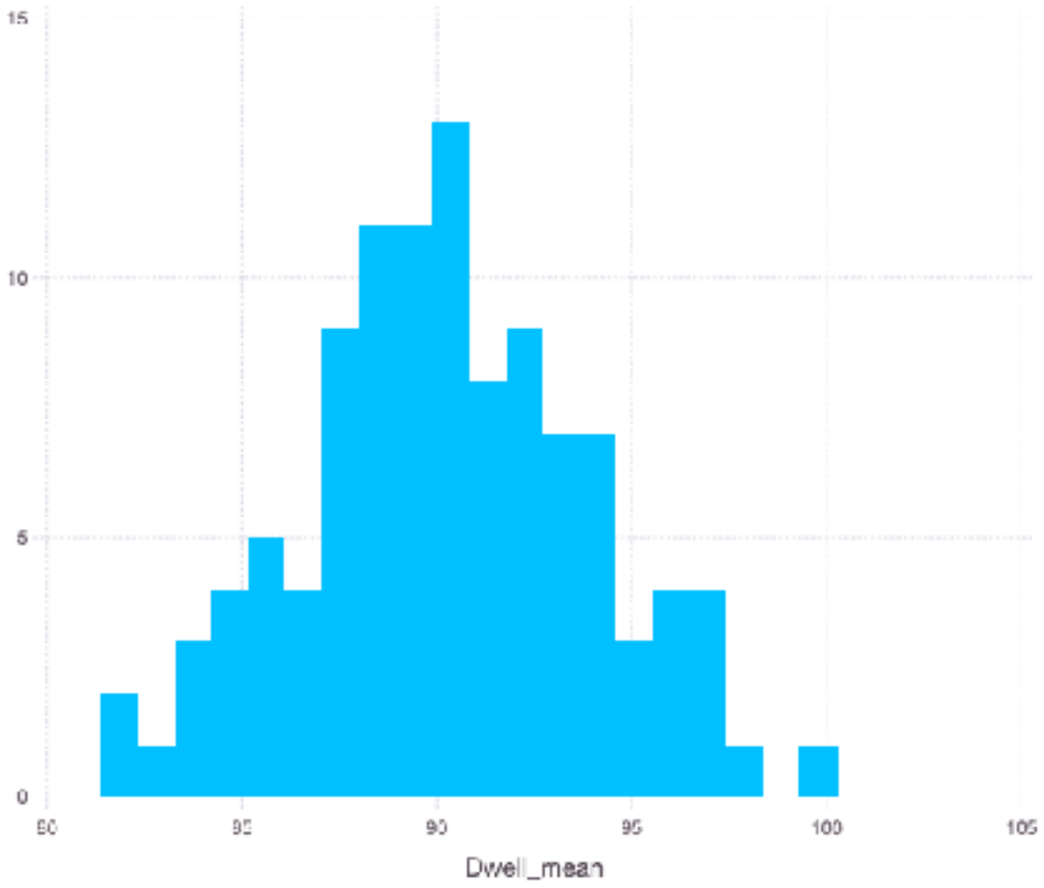


Central Limit Theorem

plot(daily_dwell, x="Dwell_mean", Geom.histogram(bincount=20))



Week Days



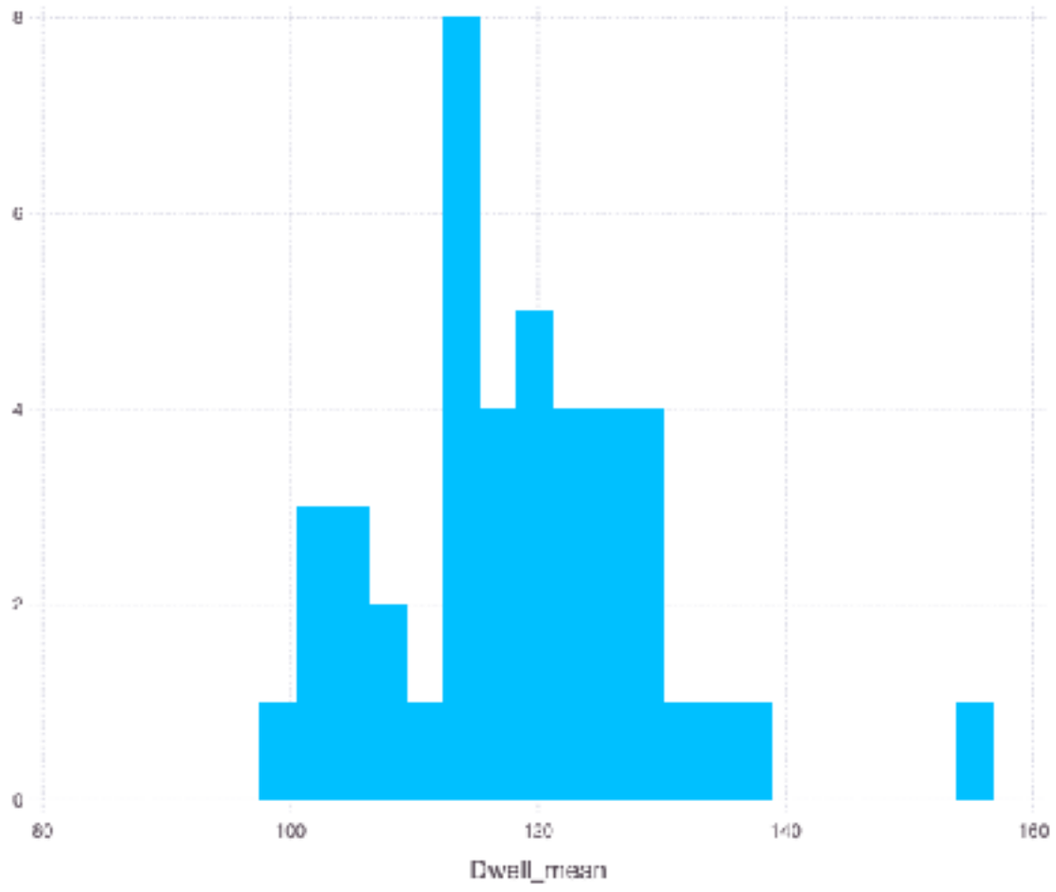
sample size = 107

mean = 90.2

std = 3.7

CI of mean p = 0.05 (115,122)

Weekends



sample size = 107

mean = 118.3

std = 11.0

CI of mean p = 0.05 (89.5 ,90.9)

Second Example - Admission to Grad School

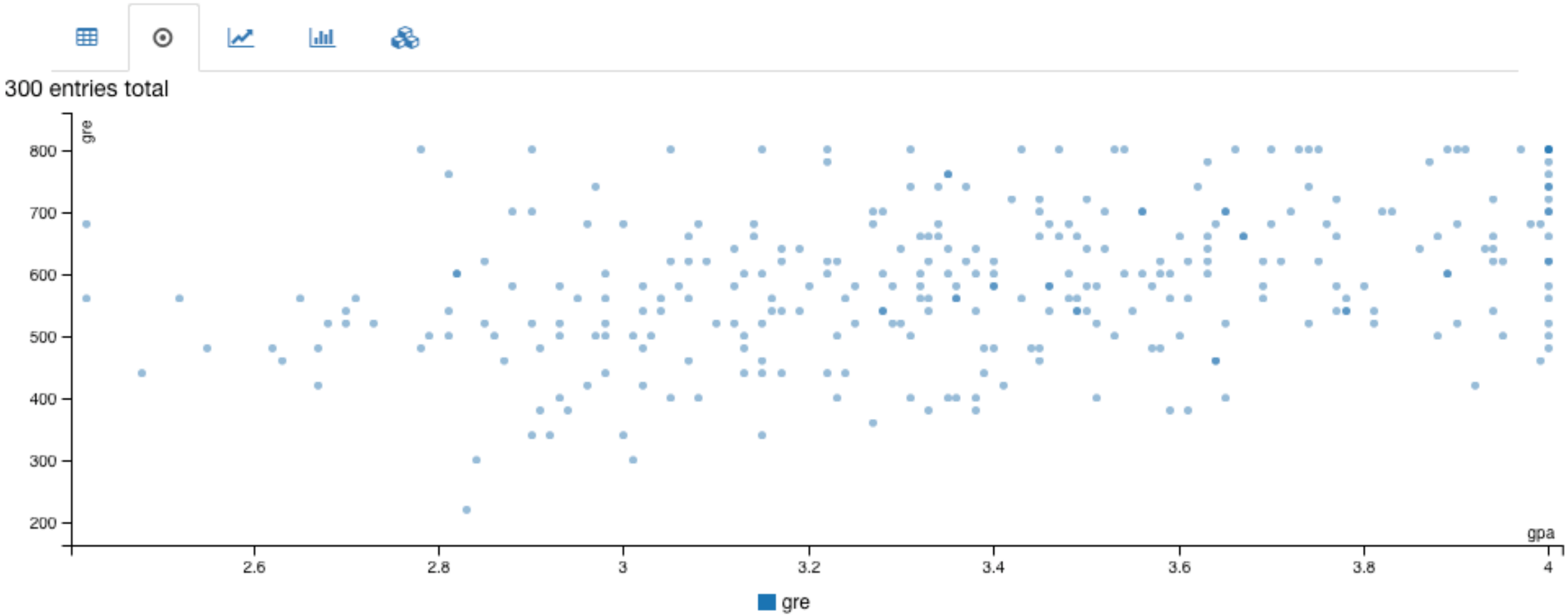
Data from http://www.ats.ucla.edu/stat/data/binary.csv
Analysis using R: https://stats.idre.ucla.edu/r/dae/logit-regression/

Given the data build a model of admissions

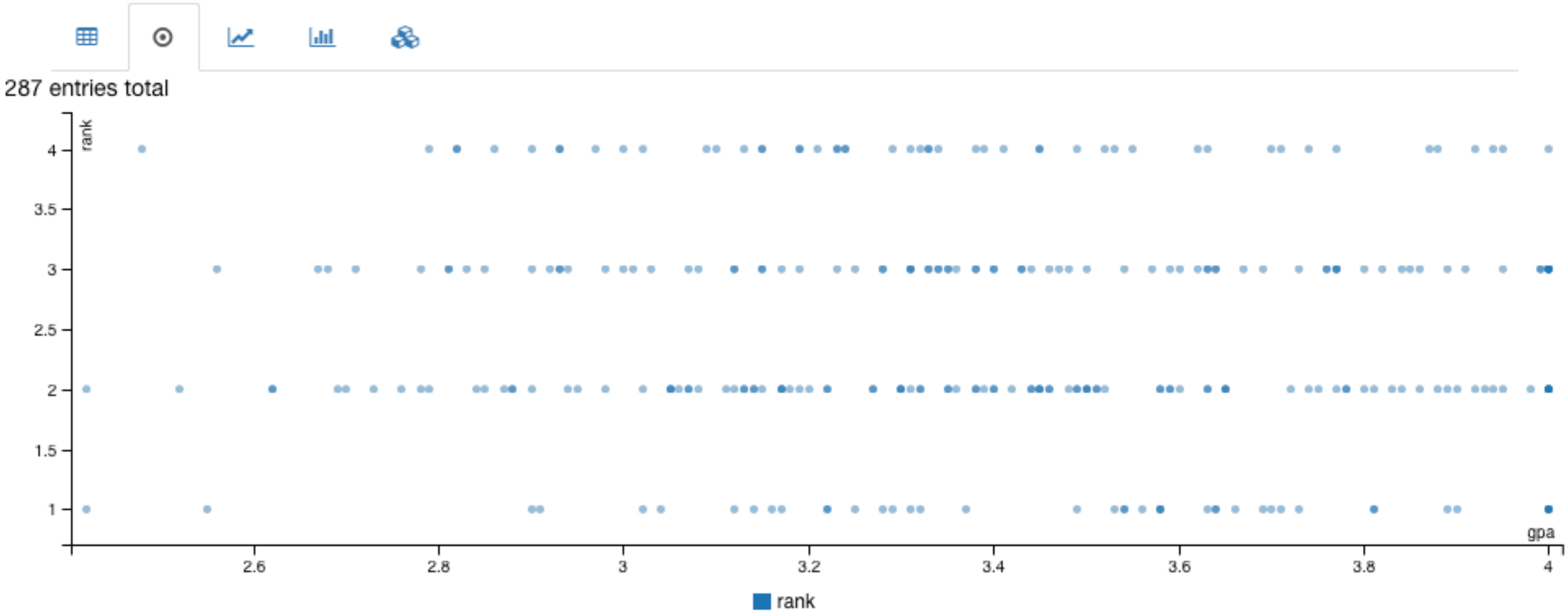
admit - dependent variable, 1 = admit
gre, gpa
rank - ranking of school student attended, 1= highest rank, 4 lowest

admit	gre	gpa	rank
0	220	2.83	3
0	340	3.15	3
0	360	3.27	3
0	380	2.91	4
0	380	3.33	4
0	380	3.38	2
0	380	3.43	3
0	380	3.59	4
0	380	3.61	3

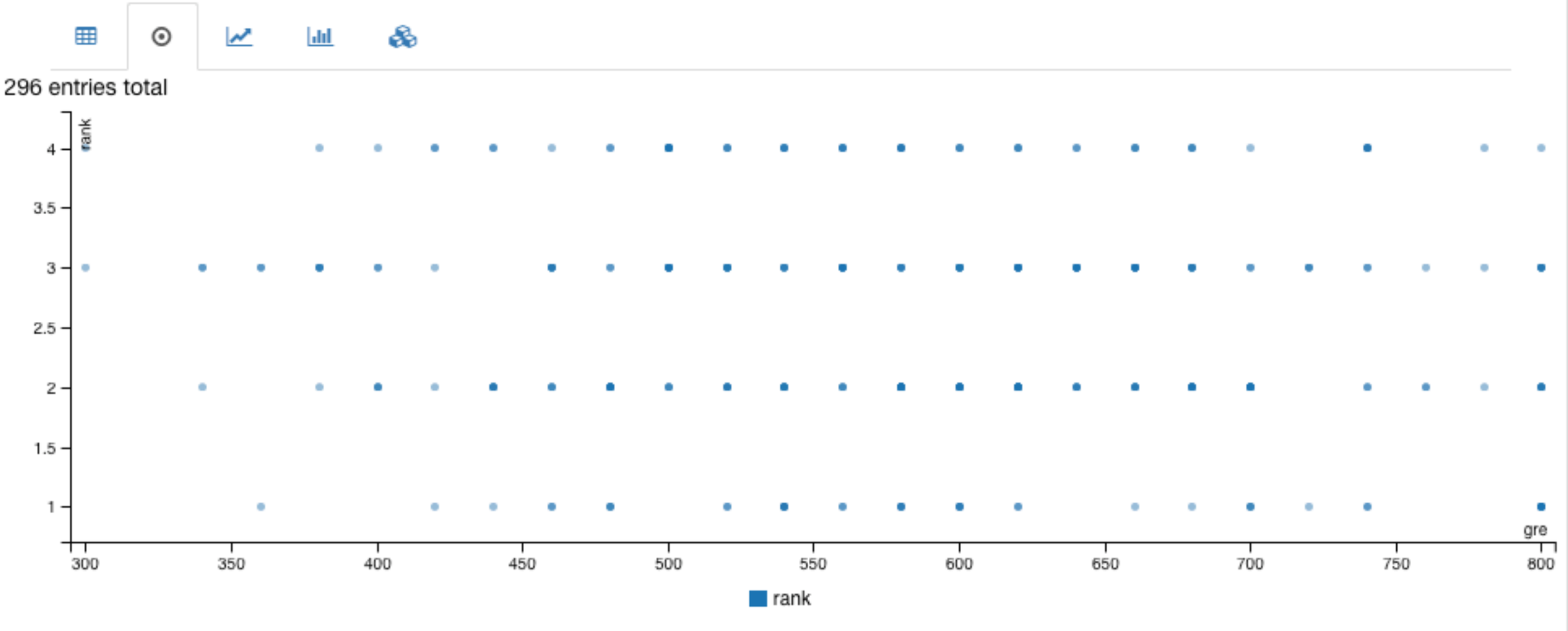
GPA vs GRE


300 entries total
gre
800
700
600
500
400
300
200
2.6
2.8
3
3.2
3.4
3.6
3.8
4
gpa
gre

Rank & GPA


287 entries total
rank
4
3.5
3
2.5
2
1.5
1
gpa
2.6
2.8
3
3.2
3.4
3.6
3.8
4
rank

GRE & Rank


296 entries total
rank
4
3.5
3
2.5
2
1.5
1
300
350
400
450
500
550
600
650
700
750
800
gre
rank

R Datasets - Academic Salaries

Table 1

	rank	discipline	yrs.since.phd	yrs.service	sex	salary
1	Prof	B	19	18	Male	139750
2	Prof	B	20	16	Male	173200
3	AsstProf	B	4	3	Male	79750
4	Prof	B	45	39	Male	115000
5	Prof	B	40	41	Male	141500
6	AssocProf	B	6	6	Male	97000
7	Prof	B	30	23	Male	175000
8	Prof	B	45	45	Male	147765
9	Prof	B	21	20	Male	119250

Jupyter Apache Toree

```
import org.apache.spark.sql.types.{StructField, StructType, StringType, DoubleType, IntegerType

val schema = new StructType(Array(
  new StructField("id", StringType, true),
  new StructField("rank", StringType, true),
  new StructField("discipline", StringType, true),
  new StructField("yrsSincePhD", IntegerType, true),
  new StructField("yrsService", IntegerType, true),
  new StructField("sex", StringType, true),
  new StructField("salary", IntegerType, true)))

val data = spark.read.format("csv").schema(schema).option("header",true).
  load("/Java/spark-notebook-0.9.0/notebooks/lectureExamples/salaries/Salaries.csv")
data.show
```

```
+---+--------+----------+-----------+----------+------+------+
| id|    rank|discipline|yrsSincePhD|yrsService|   sex|salary|
+---+--------+----------+-----------+----------+------+------+
|  1|    Prof|         B|         19|        18|  Male|139750|
|  2|    Prof|         B|         20|        16|  Male|173200|
|  3|AsstProf|         B|          4|         3|  Male| 79750|
|  4|    Prof|         B|         45|        39|  Male|115000|
|  5|    Prof|         B|         40|        41|  Male|141500|
```

Spark-Notebook

```
import org.apache.spark.sql.types.{StructField, StructType, StringType, DoubleType, IntegerTyp

val schema = new StructType(Array(same as previous slide))

val data = spark.read.format("csv").schema(schema).option("header",true).
  load("/Java/spark-notebook-0.9.0/notebooks/lectureExamples/salaries/Salaries.csv")

data
```

1 >>

id	rank	discipline	yrsSincePhD	yrsService	sex	salary
"1"	"Prof"	"B"	19	18	"Male"	139750
"2"	"Prof"	"B"	20	16	"Male"	173200
"3"	"AsstProf"	"B"	4	3	"Male"	79750
"4"	"Prof"	"B"	45	39	"Male"	115000
"5"	"Prof"	"B"	40	41	"Male"	141500
"6"	"AssocProf"	"B"	6	6	"Male"	97000
"7"	"Prof"	"B"	30	23	"Male"	175000
"8"	"Prof"	"B"	45	45	"Male"	147765
"9"	"Prof"	"B"	21	20	"Male"	119250
"10"	"Prof"	"B"	18	18	"Female"	129000
"11"	"AssocProf"	"B"	12	8	"Male"	119800

Spark-Notebook

data.collect

397 entries total

Show: 50

Search:

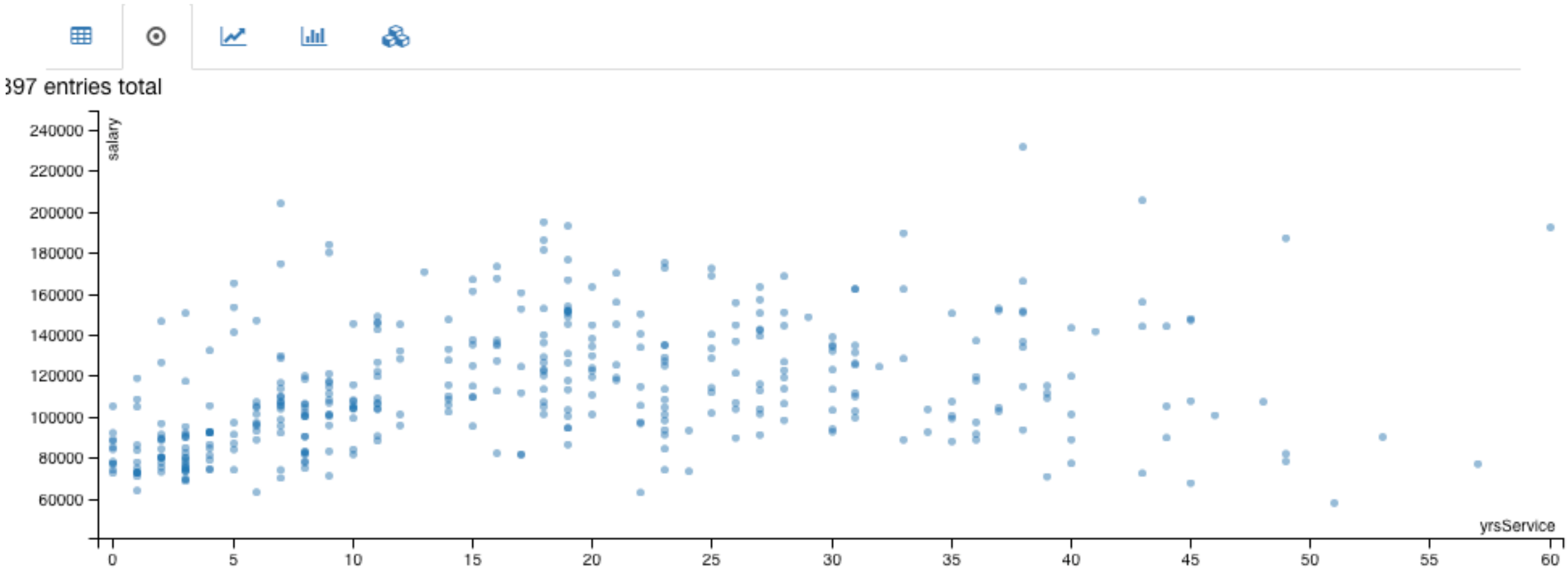
id	rank	discipline	yrsSincePhD	yrsService	sex	salary ▲
183	AssocProf	B	8	8	Male	100000
141	AssocProf	A	14	8	Male	100102
73	Prof	B	29	19	Male	100131
59	AssocProf	B	10	9	Male	100135

397 entries total

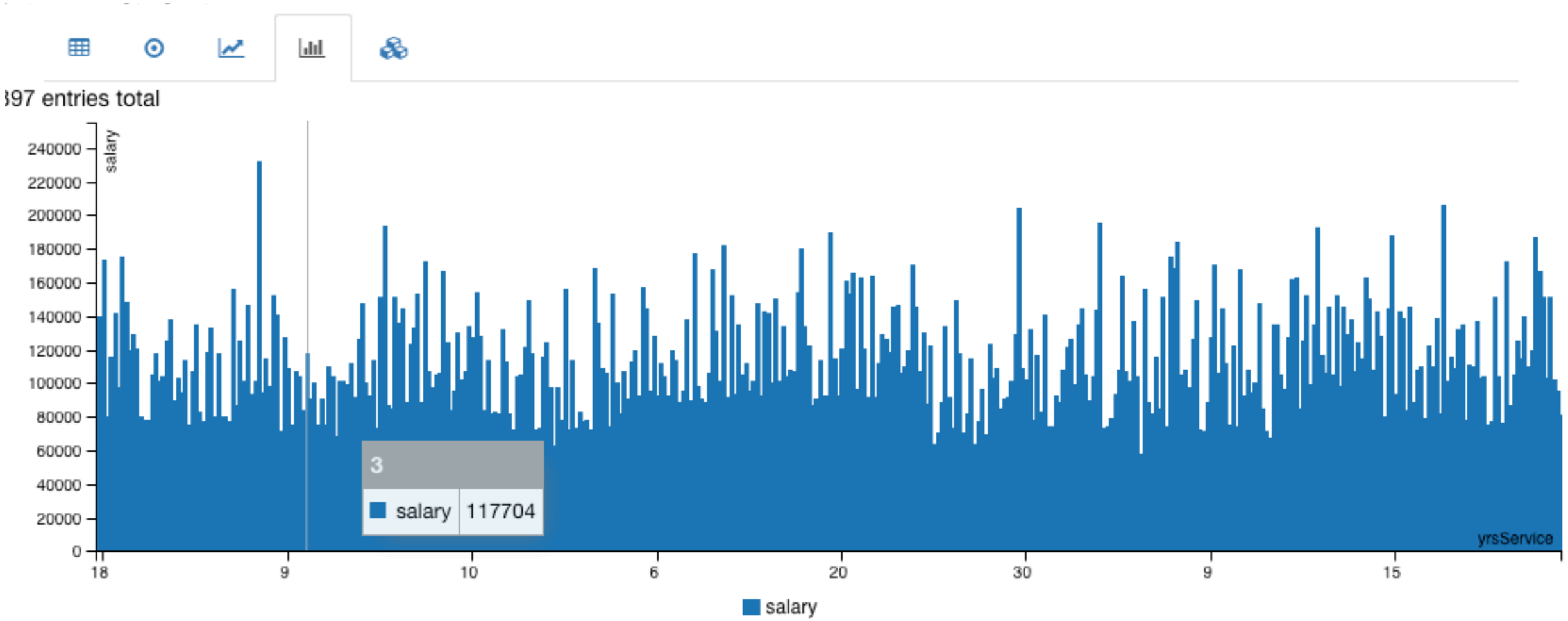
Show: 50

id	rank	discipline	yrsSincePhD ▲	yrsService	sex	salary
50	AsstProf	B	1	1	Male	70768
13	AsstProf	B	1	1	Male	77700
158	AsstProf	B	1	0	Male	88000
165	AsstProf	B	1	0	Male	88795
59	AssocProf	B	10	9	Male	100135

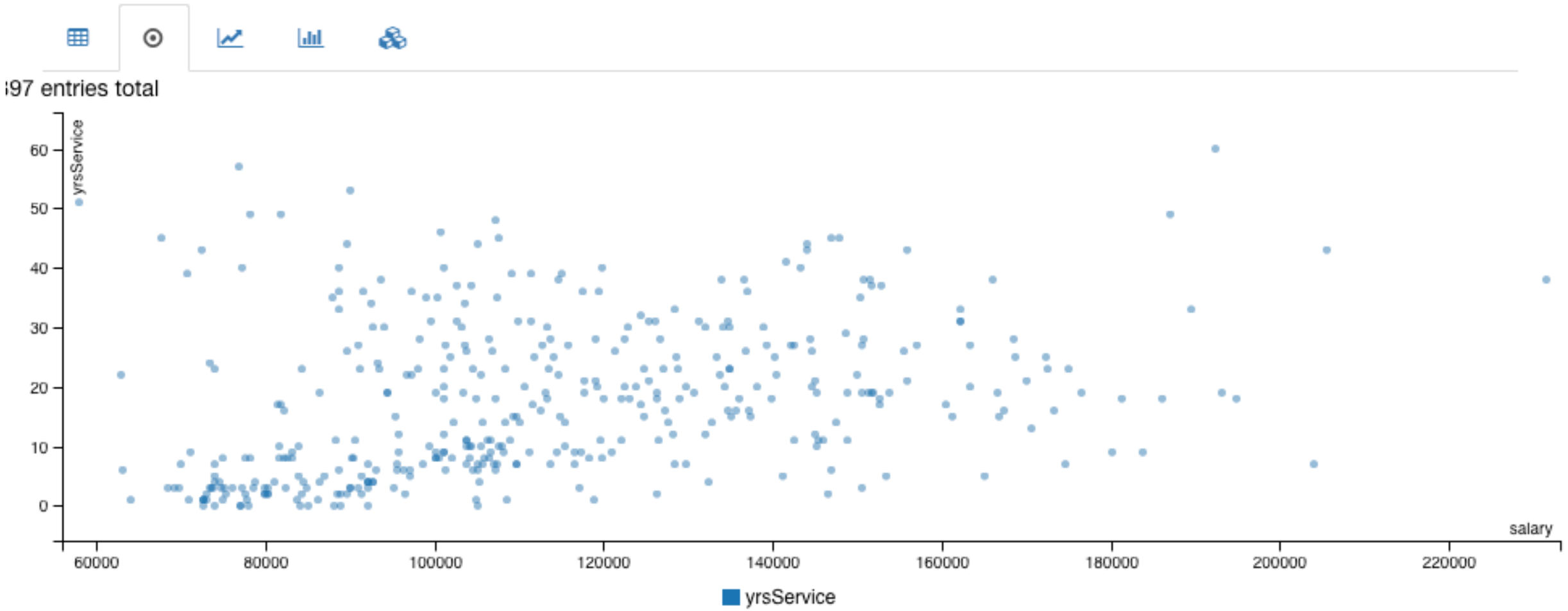
data.select("yrsService","salary").collect



data.select("yrsService","salary").collect

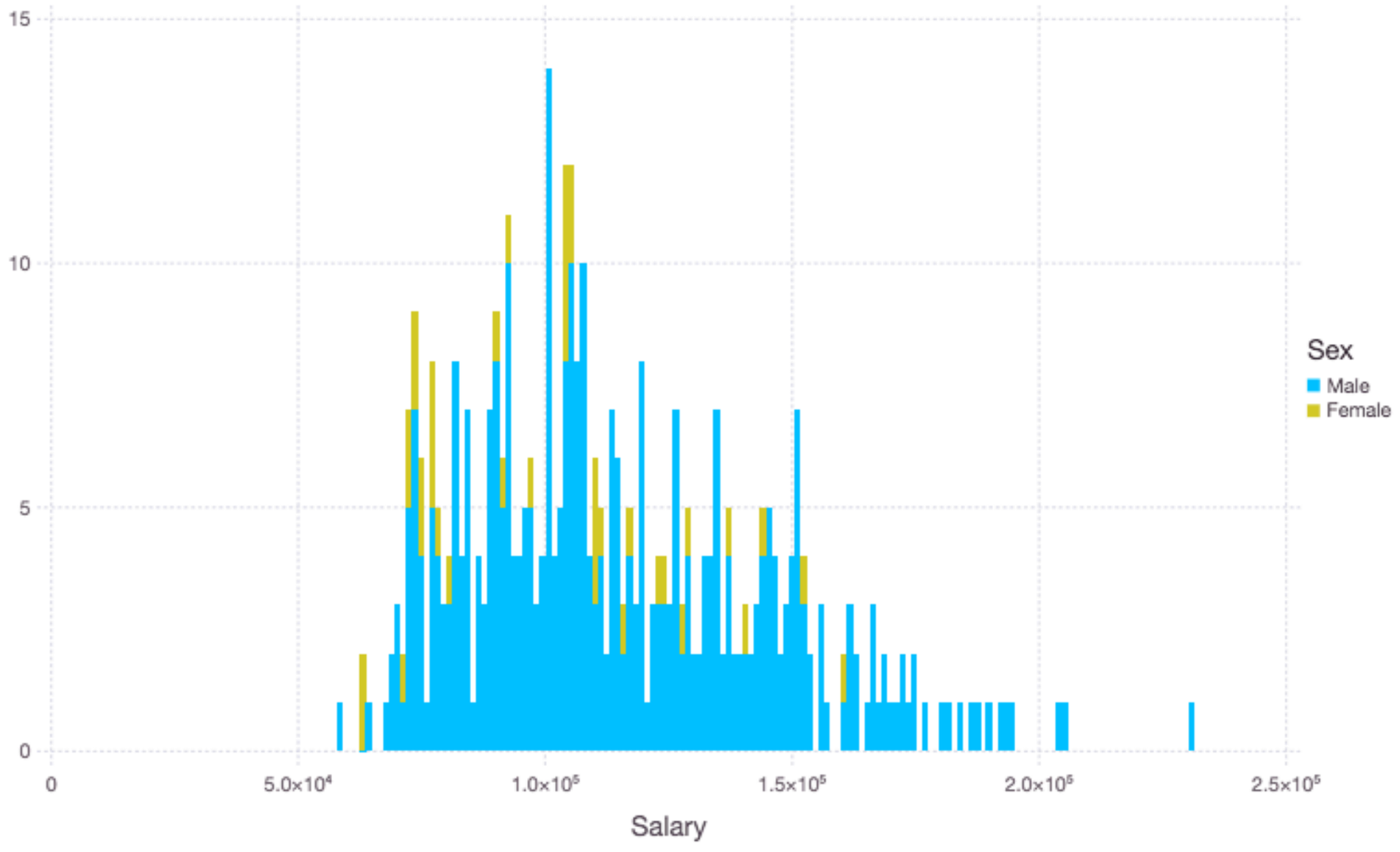


data.select("salary","yrsService").collect



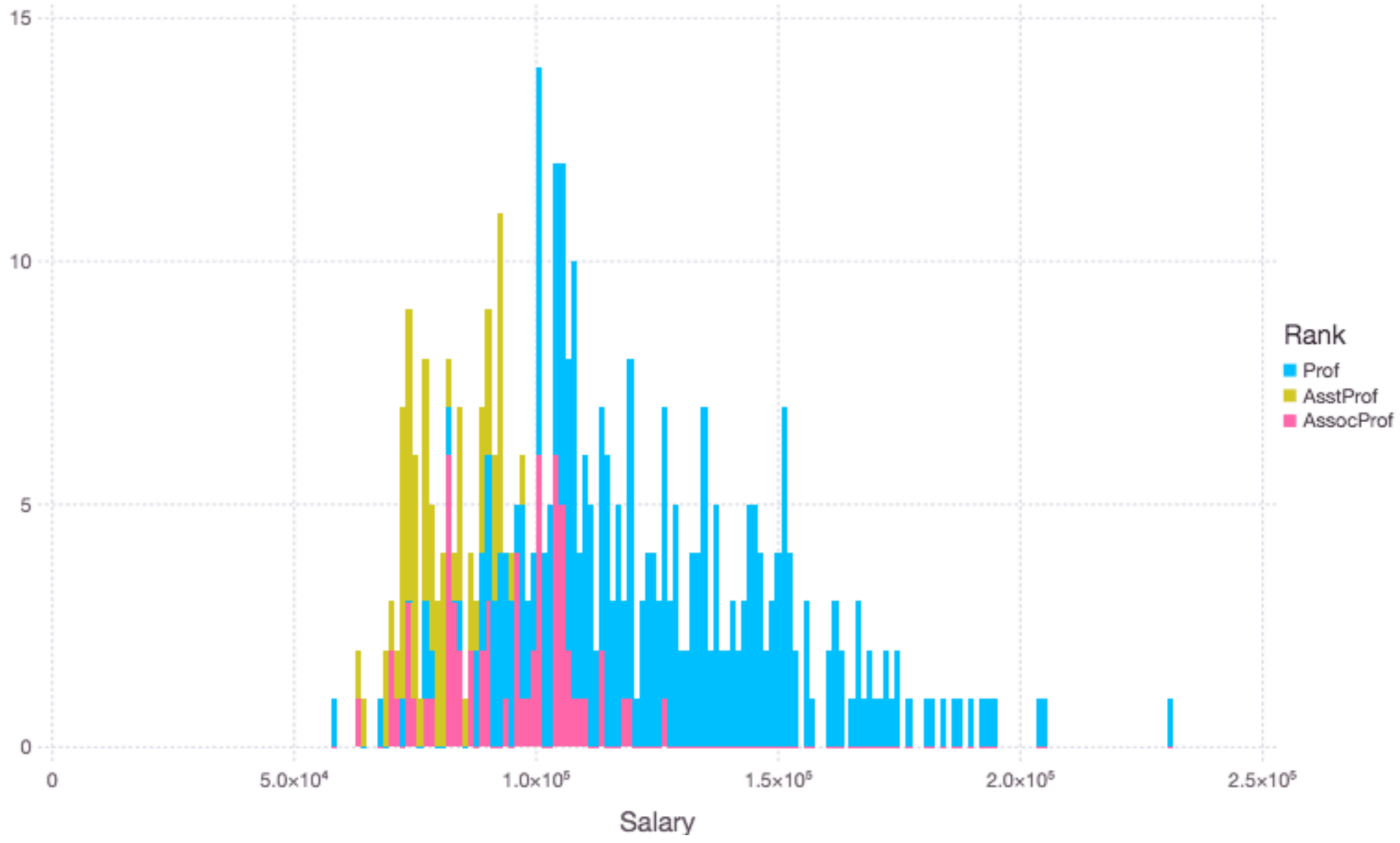
Salary & Sex

plot(dataset("car", "Salaries"), x="Salary", color="Sex", Geom.histogram)



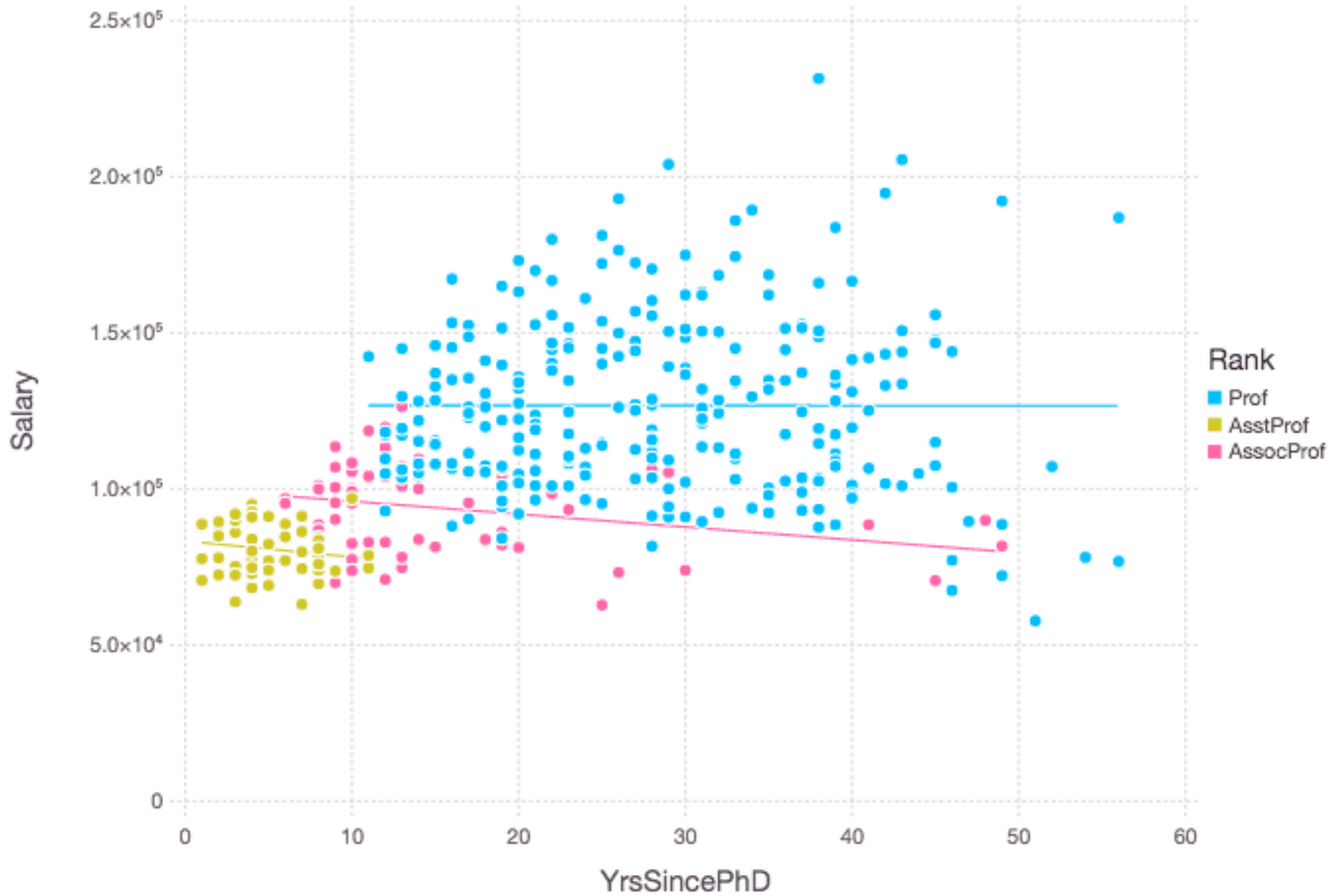
Salary & Rank

plot(dataset("car", "Salaries"), x="Salary", color="Rank", Geom.histogram)



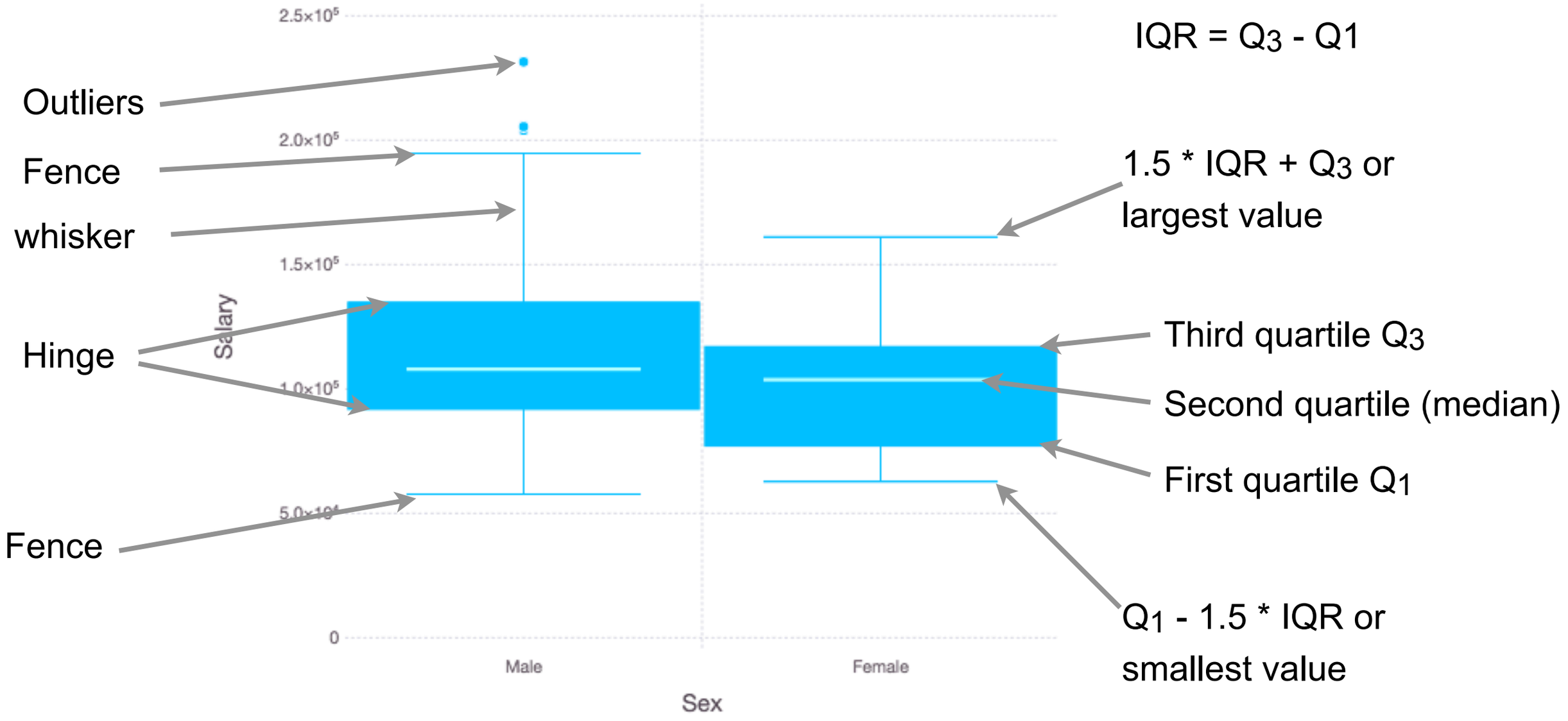
Scatter Plot: Salary-Years Colored by Rank

```
plot(dataset("car", "Salaries"), y="Salary", x="YrsSincePhD", color="Rank",
    Geom.point,
    Geom.smooth(method=:lm))
```



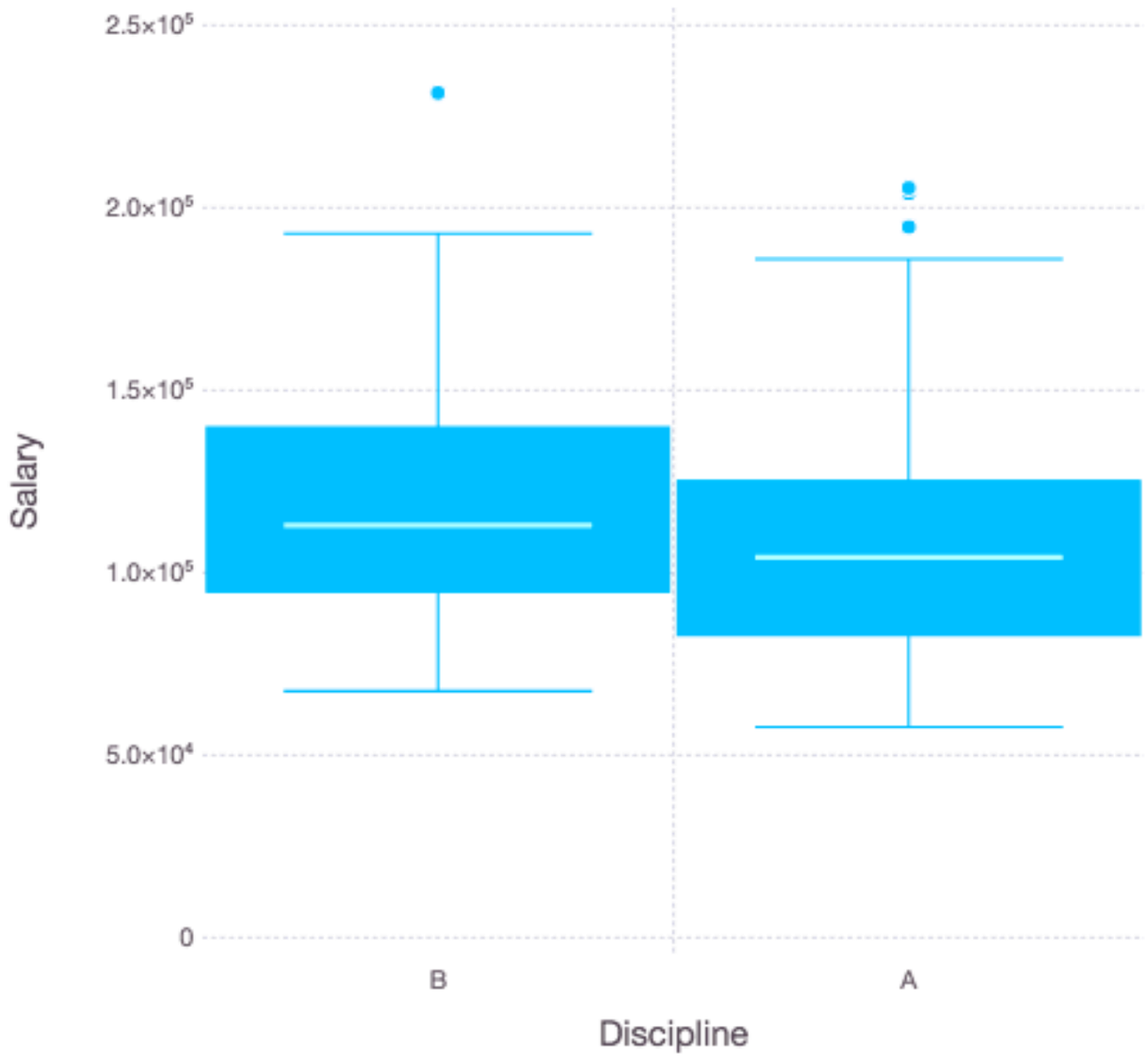
Box Plots (Tukey Method)

plot(dataset("car", "Salaries"), y="Salary", x="Sex", Geom.boxplot)



Salary by Discipline

plot(dataset("car", "Salaries"), y="Salary", x="Discipline",Geom.boxplot)

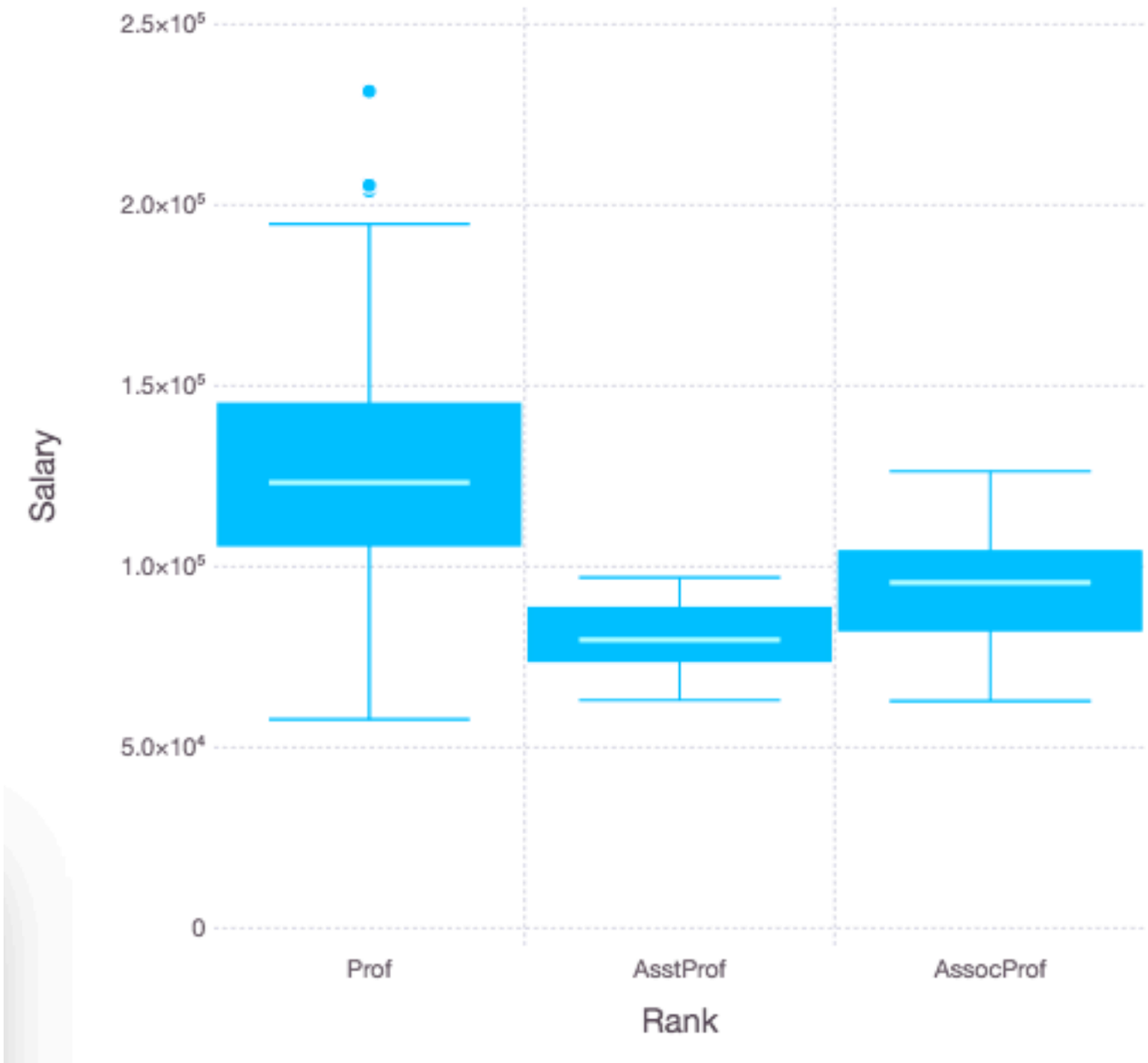


A = Theoretical

B = Applied

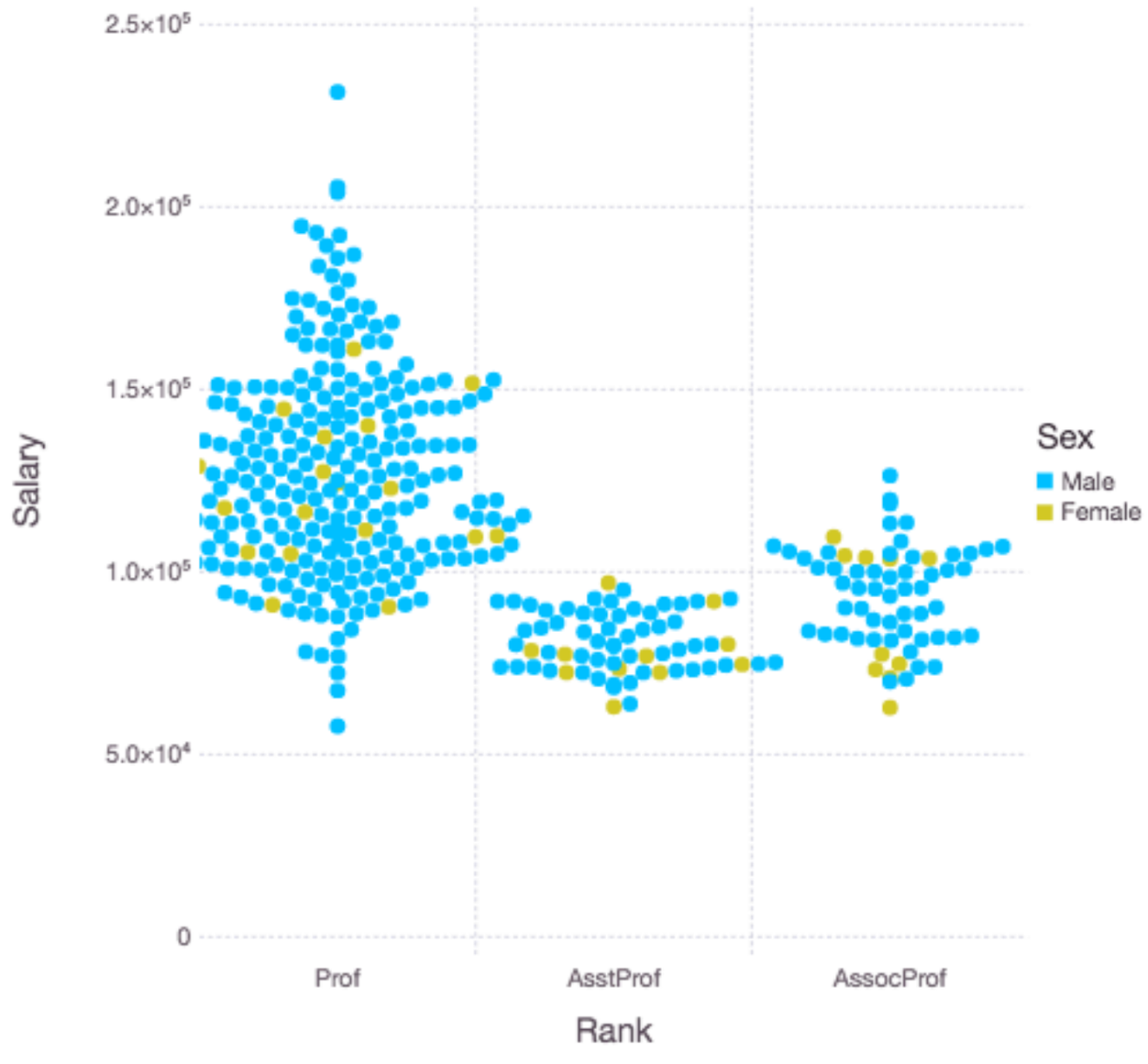
Salary by Rank

plot(dataset("car", "Salaries"), y="Salary", x="Rank",Geom.boxplot)



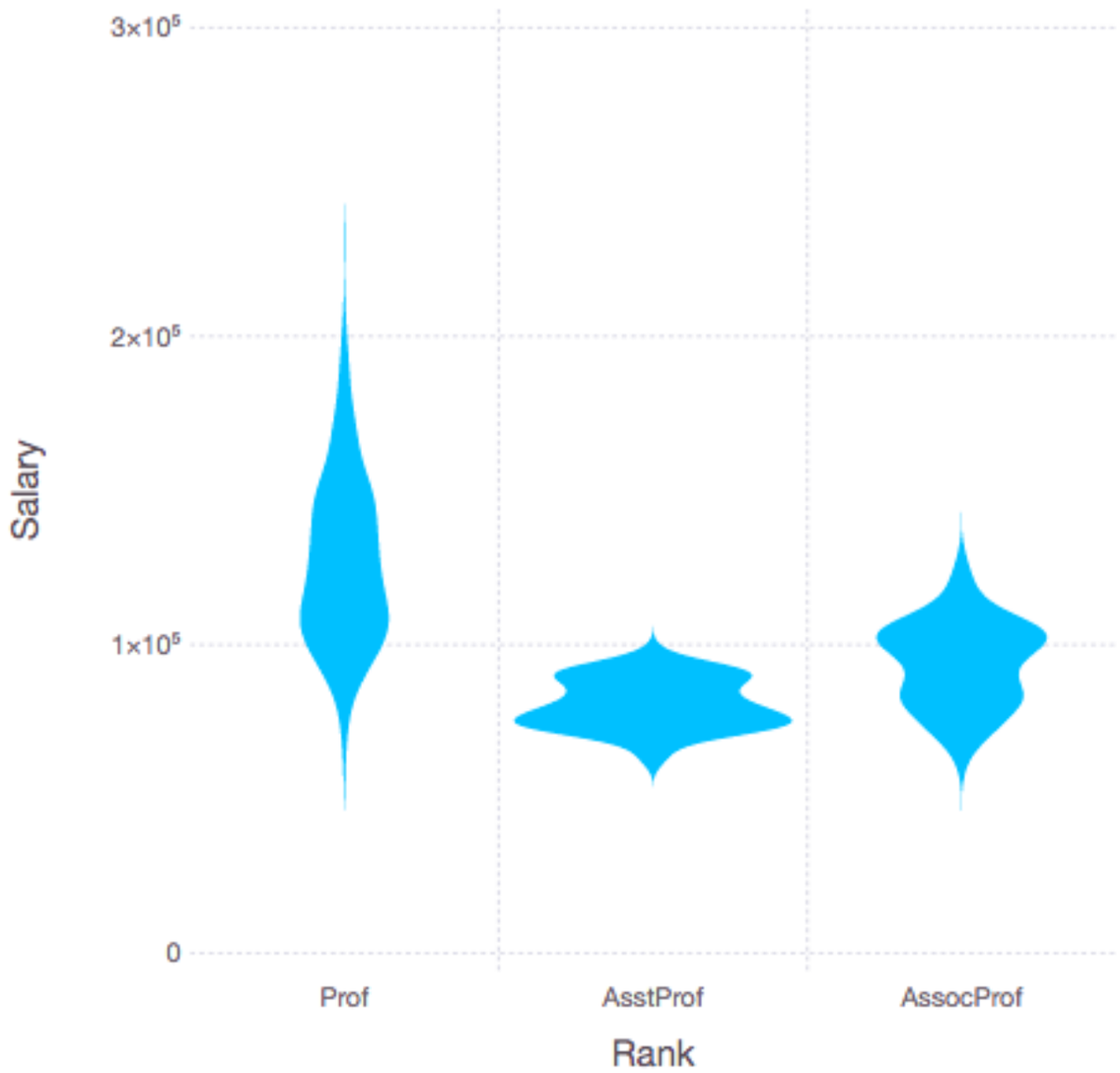
Beeswarm: Salary by Rank with Sex

plot(dataset("car", "Salaries"), x="Rank", y="Salary",color="Sex",Geom.beeswarm)



Violin Plot: Salary by Rank

plot(dataset("car", "Salaries"), x="Rank", y="Salary",Geom.violin)



Iris Dataset

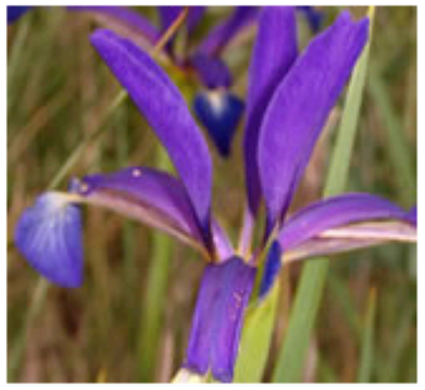
UCI Machine Learning Repository

Center for Machine Learning and Intelligent Systems

Iris Data Set

Download: Data Folder, Data Set Description

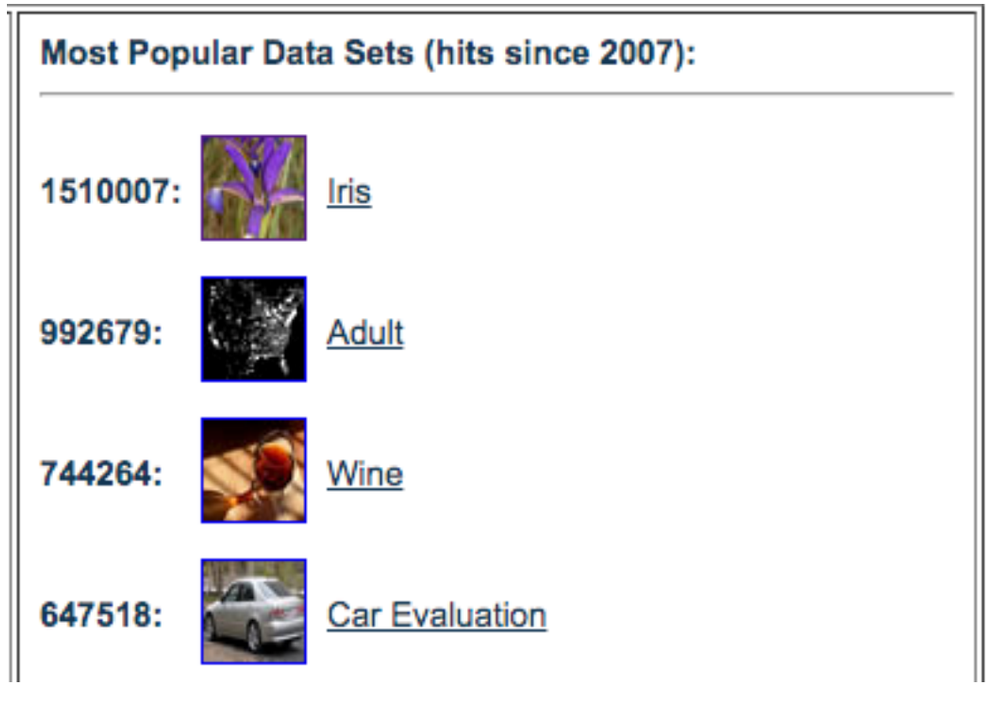
Abstract: Famous database; from Fisher, 1936



Data Set Characteristics:	Multivariate	**Number of Instances:**	150	**Area:**	Life
Attribute Characteristics:	Real	**Number of Attributes:**	4	**Date Donated**	1988-07-01
Associated Tasks:	Classification	**Missing Values?**	No	**Number of Web Hits:**	1510039

Iris Dataset

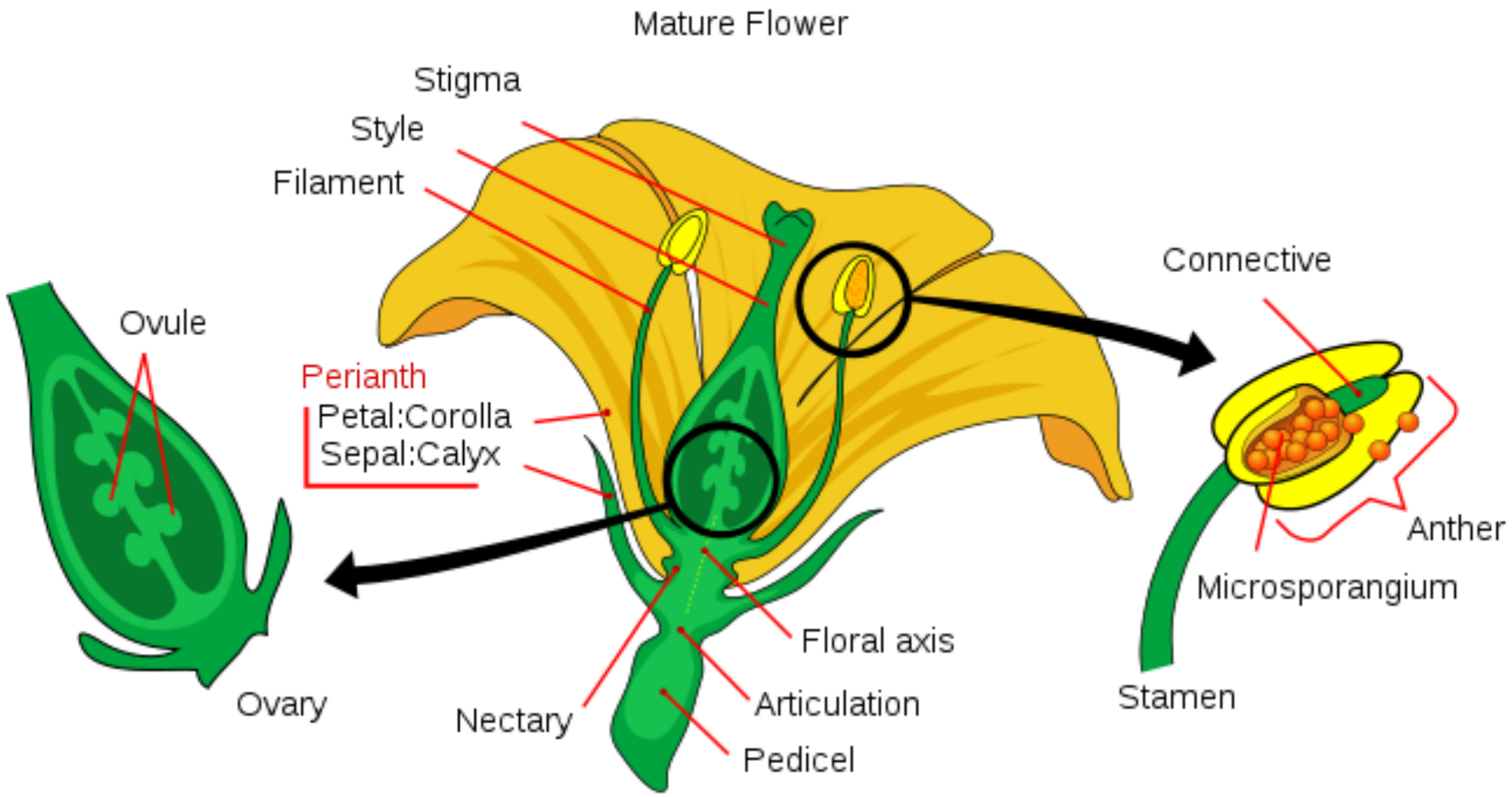
4.8,3.0,1.4,0.3,Iris-setosa
5.1,3.8,1.6,0.2,Iris-setosa
4.6,3.2,1.4,0.2,Iris-setosa
5.3,3.7,1.5,0.2,Iris-setosa
5.0,3.3,1.4,0.2,Iris-setosa
7.0,3.2,4.7,1.4,Iris-versicolor
6.4,3.2,4.5,1.5,Iris-versicolor
6.9,3.1,4.9,1.5,Iris-versicolor
5.5,2.3,4.0,1.3,Iris-versicolor
6.5,2.8,4.6,1.5,Iris-versicolor



Columns

1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm
5. class:
   -- Iris Setosa
   -- Iris Versicolour
   -- Iris Virginica

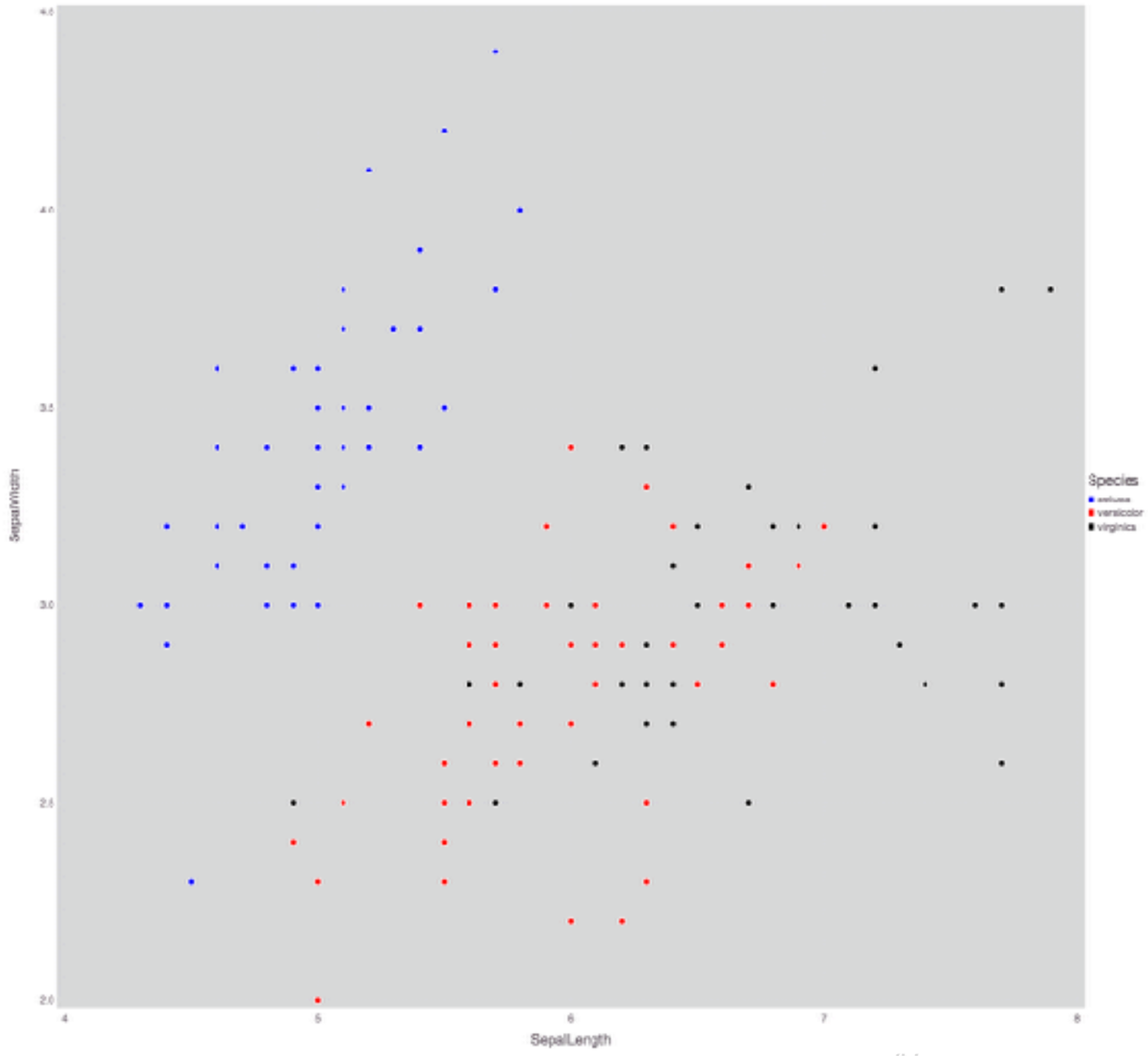
Sepal & Petal


Mature Flower
Stigma
Style
Filament
Ovule
Perianth
Petal:Corolla
Sepal:Calyx
Connective
Anther
Microsporangium
Floral axis
Articulation
Pedicel
Nectary
Ovary
Stamen

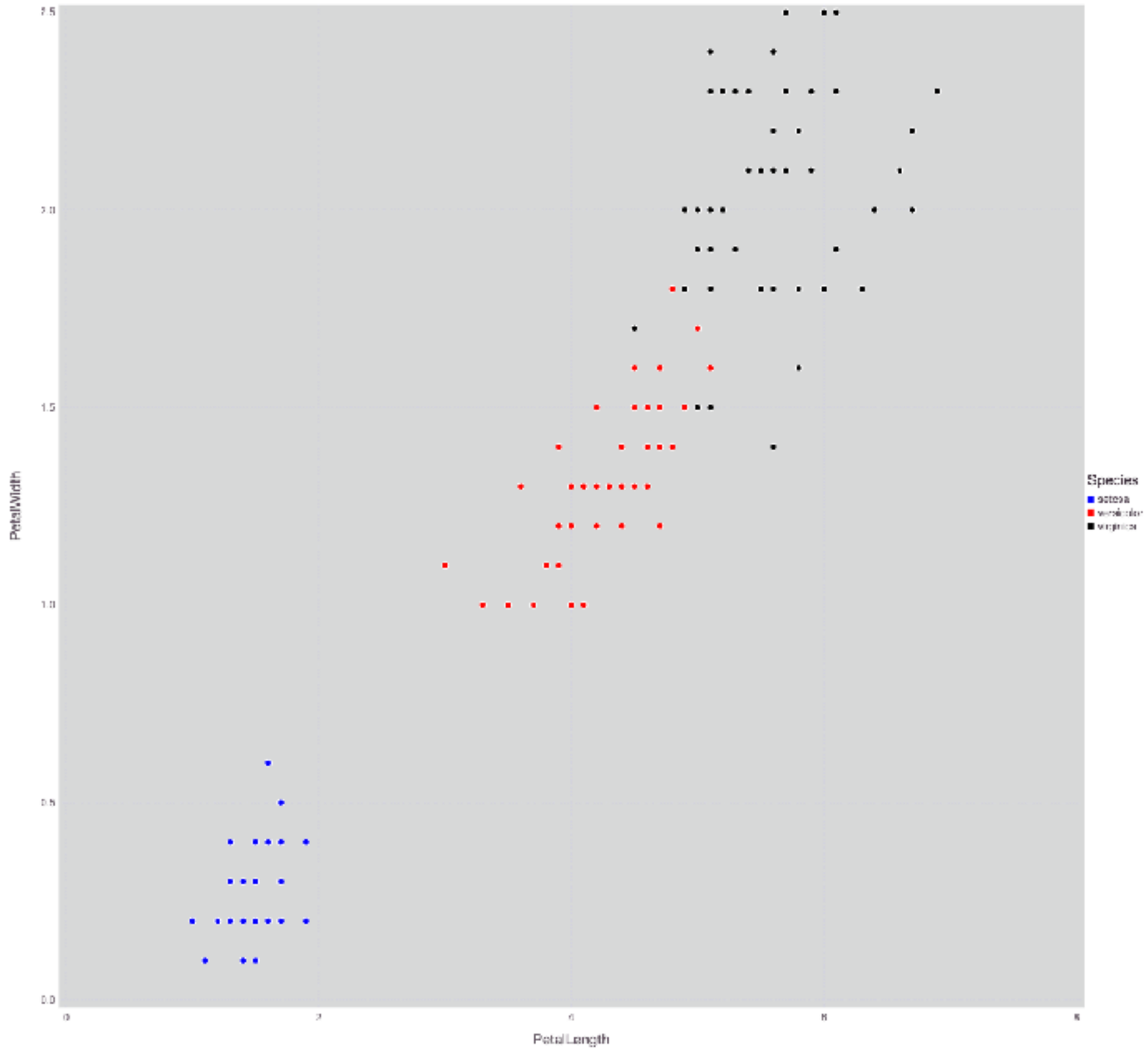
Basic Question

Given sepal & petal dimensions which iris is it?

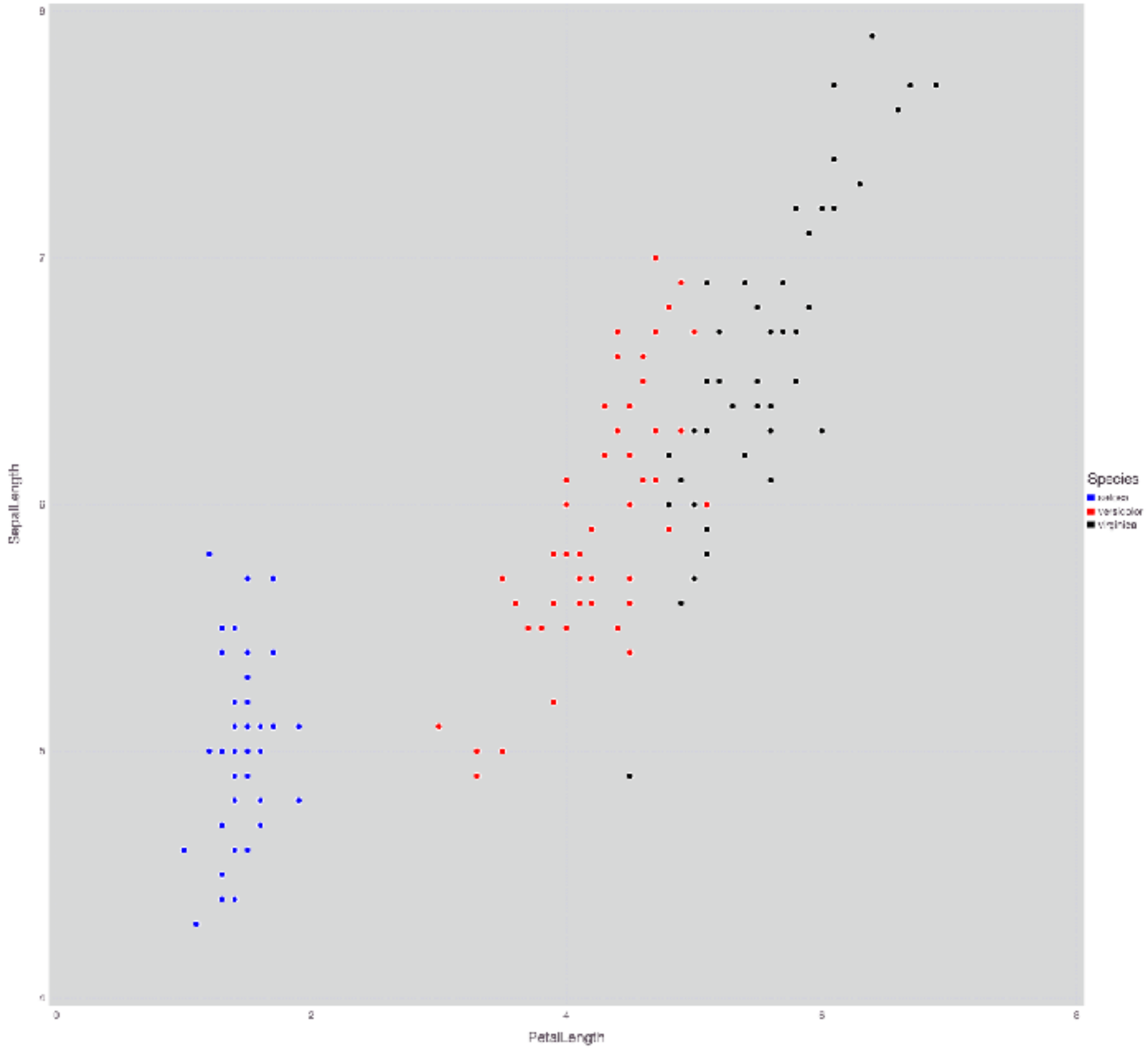
Sepal Width vs length



Petal Width vs Length



Sepal vs Petal Length



Kaggle Notebooks & Iris Dataset

Decision Boundaries visualized via Python & Plotly

Kaggle

http://tinyurl.com/yb9bgkue

Python data visualizations on the Iris dataset

http://tinyurl.com/y9j3qv4j

Thursday Tasks

Classify Iris dataset using Scala & Spark

- Decision Trees
- Random forest classifier
- Logisitic Regression

Use RegressionEvaluator & MulticlassClassificationEvaluator