USC Marshall School of Business

DSO 545: STATISTICAL COMPUTING AND DATA VISUALIZATION

Instructor:	Francis Pereira, Ph.D.
Office:	BRI 401 M
Office Hours:	Tuesdays 12 pm – 1 pm
	By Zoom
Email:	pereira@marshall.usc.edu
Instructional Assistant:	TBA

Why Study Data Science and Statistics?

- Turn data into information.
  - Your job as managers is to make decisions.
  - You need to make the most informed decisions that you can.
- Decision making under uncertainty.
  - Most of your decisions are based on guesses, rather than “facts.”
  - You will learn how to make the “best” guess you can as well as how to measure the accuracy of your guesses.

Course Description

Students will learn how to utilize Python to answer data science problems in marketing, finance, operations, and human resources. Utilizing Python as a tool in our data science process, we will:

- Read different types of data (structured and unstructured)
- Clean, manipulate, and aggregate the data into a useful format
- Explore the data using numerical summaries
- Explore the data using data visualization
- Explore the data using clustering techniques
- Run A/B tests
- Run regression analysis
- Use and create Python data structures, control statements, functions, and classes
- Scrape data from the web
- Create web interactive dashboards to present your results to different business stakeholders

Course Objectives

Understand

- Interpret business problems using exploratory data analysis (EDA)
- Interpret and communicate the outcomes of the EDA process

Apply

- Clean and prepare datasets for analysis
- Produce graphs that follow the grammar of graphics to support the EDA process
- Produce interactive dashboards to support the EDA process
- Access data through different sources (databases, cloud, and web scrapping)

Analyze

- Analyze business case studies using data science tools to make decisions in marketing, finance, human resources, and operations

Evaluate

- Recommend business strategies based on evidence coming from data

Create

- Prepare a data project management plan for real-world data science problems
- Prepare data science solutions for real-world business problems

Course Material

Books *(soft copy available for USC students https://libraries.usc.edu)*

1. Python for Marketing Research and Analytics. Springer 2020. (by Jason Schwarz, Chris Chapman, and Elea McDonnell Feit)

- On Blackboard

Course Assessment

Assessment	% of Grade
Homework	20%
Team Project	30%
Midterm	25%
Final Exam	25%

Use of AI Platforms

In this course, I encourage you to use artificial intelligence (AI)-powered programs to help you with assignments that indicate the permitted use of AI. You should also be aware that AI text generation tools may present incorrect information, biased responses, and incomplete analyses; thus they are not yet prepared to produce text that meets the standards of this course. To adhere to our university values, you must cite any AI-generated material (e.g., text, images, etc.) included or referenced in your work and provide the prompts used to generate the content. Using an AI tool to generate content without proper attribution will be treated as plagiarism and reported to the Office of Academic Integrity. Please review the instructions in each assignment for more details on how and when to use AI Generators for your submissions.

In this regard, please note that part or all of examinations may involve python writing code in a non-electronic medium.

Academic Integrity

The University of Southern California is a learning community committed to developing successful scholars and researchers dedicated to the pursuit of knowledge and the dissemination of ideas. Academic misconduct, which includes any act of dishonesty in the production or submission of academic work, comprises the integrity of the person who commits the act and can impugn the perceived integrity of the entire university community. It stands in opposition to the university's mission to research, educate, and contribute productively to our community and the world.

All students are expected to submit assignments that represent their own original work, and that have been prepared specifically for the course or section for which they have been submitted. You may not submit work written by others or "recycle" work prepared for other courses without obtaining written permission from the instructor(s).

Students and Disability Accommodations:

USC welcomes students with disabilities into all of the University's educational programs. The Office of Student Accessibility Services (OSAS) is responsible for the determination of appropriate accommodations for students who encounter disability-related barriers. Once a student has completed the OSAS process (registration, initial appointment, and submitted documentation) and accommodations are determined to be reasonable and appropriate, a Letter of Accommodation (LOA) will be available to generate for each course. The LOA must be given to each course instructor by the student and followed up with a discussion. This should be done as early in the semester as possible as accommodations are not retroactive. More information can be found at osas.usc.edu. You may contact OSAS at (213) 740-0776 or via email at osasfrontdesk@usc.edu.

Support Systems:

Counseling and Mental Health - (213) 740-9355 – 24/7 on call

Free and confidential mental health treatment for students, including short-term psychotherapy, group counseling, stress fitness workshops, and crisis intervention.

988 Suicide and Crisis Lifeline - 988 for both calls and text messages – 24/7 on call

The 988 Suicide and Crisis Lifeline (formerly known as the National Suicide Prevention Lifeline) provides free and confidential emotional support to people in suicidal crisis or emotional distress 24 hours a day, 7 days a week, across the United States. The Lifeline is comprised of a national network of over 200 local crisis centers, combining custom local care and resources with national standards and best practices. The new, shorter phone number makes it easier for people to remember and access mental health crisis services (though the previous 1 (800) 273-8255 number will continue to function indefinitely) and represents a continued commitment to those in crisis.

Relationship and Sexual Violence Prevention Services (RSVP) - (213) 740-9355(WELL) – 24/7 on call

Free and confidential therapy services, workshops, and training for situations related to gender- and power-based harm (including sexual assault, intimate partner violence, and stalking).

Outline

- Why do we study data science and statistics?
- Data Science Applications
- Lab 1: Overview of Python

Data Science and statistics can help us answer

- What is the right price of an iPad?
- Which customers are interested in a tablet? How much are they willing to pay?
- Why does a shopper choose a particular box of cereal?
- Which customers are we more likely to lose?
- What makes our employees satisfied?
- What are the characteristic of a song that is a top charter?

Statistics can help us model variations

- There are different groups of customers, and each group differ from the on different dimensions: e.g. age, gender, salary, education, etc
- Variation refers to differences among products, people, entities
- Statistics provides us with tools for handling these variations

Statistics can help us model variations

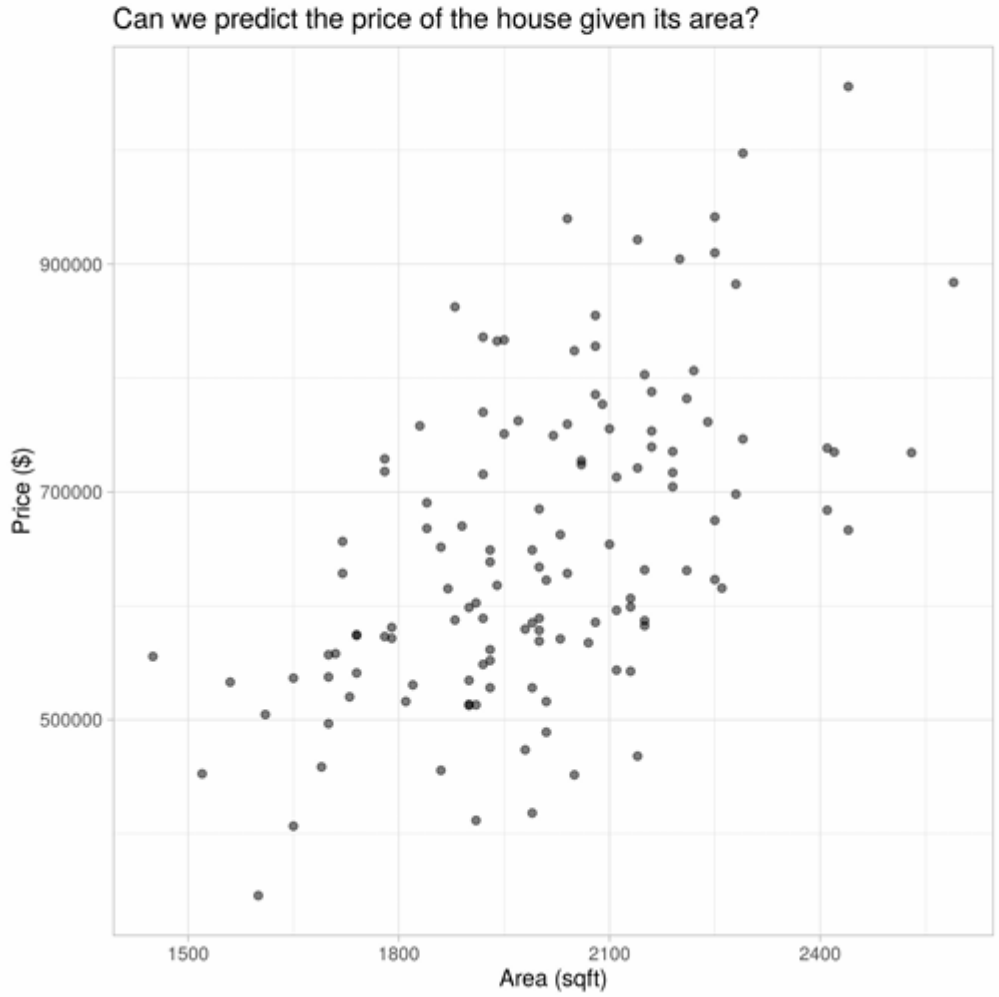
- Statistics gives us the tools to identify the signal from the noise among these variations in the data
  - The signal is a pattern in the data (explained variation)
  - The noise is the unexplained variation in the data
- These patterns are systematic features in the data, and are captured by a statistical model

Example 1: House prices

Do you see any patterns in the house prices data?

price	squarefeet	bedrooms	bathrooms	brick	neighborhood
571500	1790	2	2	No	East
571000	2030	4	2	No	East
574000	1740	3	2	No	East
473500	1980	3	2	No	East
599000	2130	3	3	No	East
573000	1780	3	2	No	North
758000	1830	3	3	Yes	West
753500	2160	4	2	No	West
596000	2110	4	2	No	East
520000	1730	3	3	No	East

Do we see any pattern?



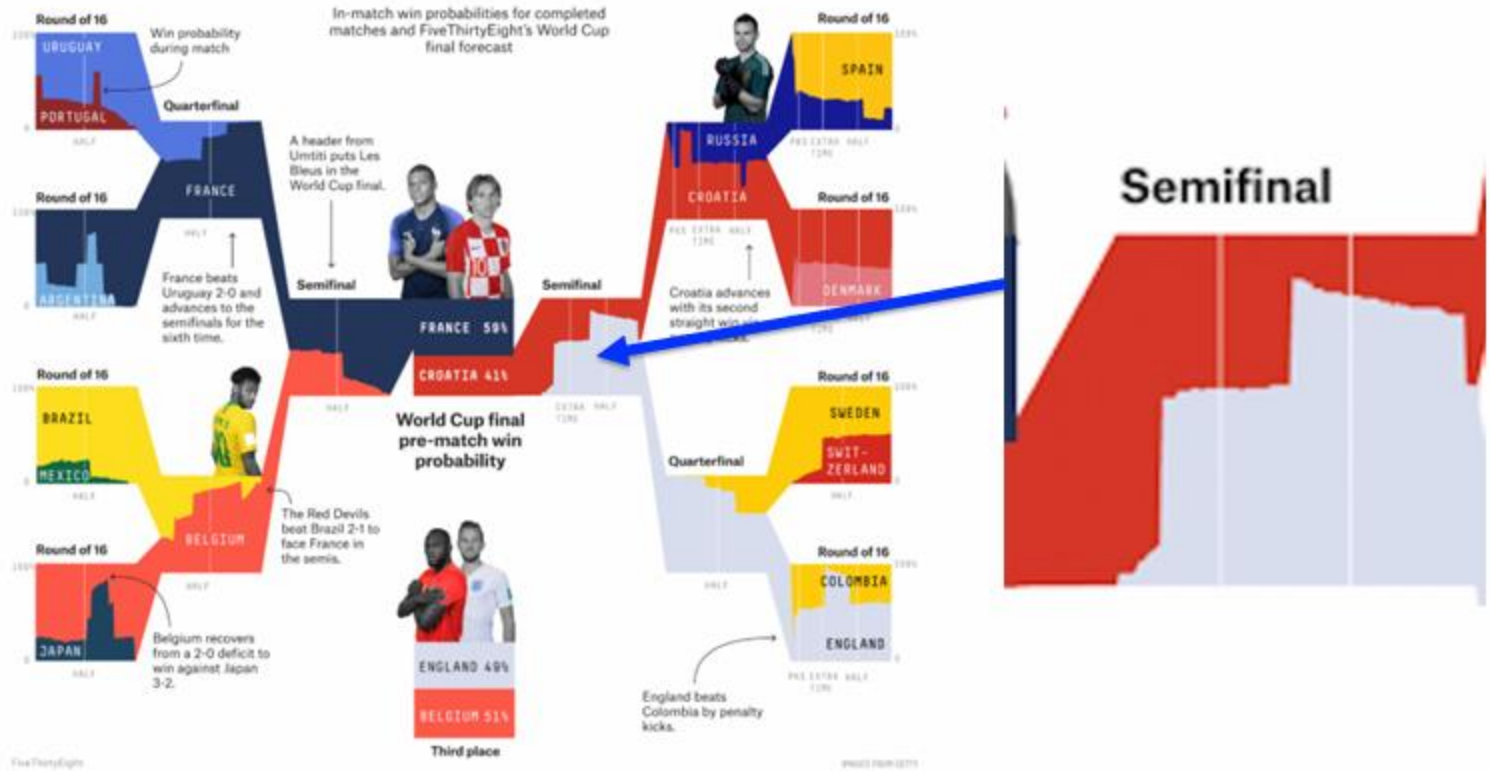
Statistics helps us detect a linear pattern in this data

Do you see any pattern?

Why Study Data Science and Statistics?

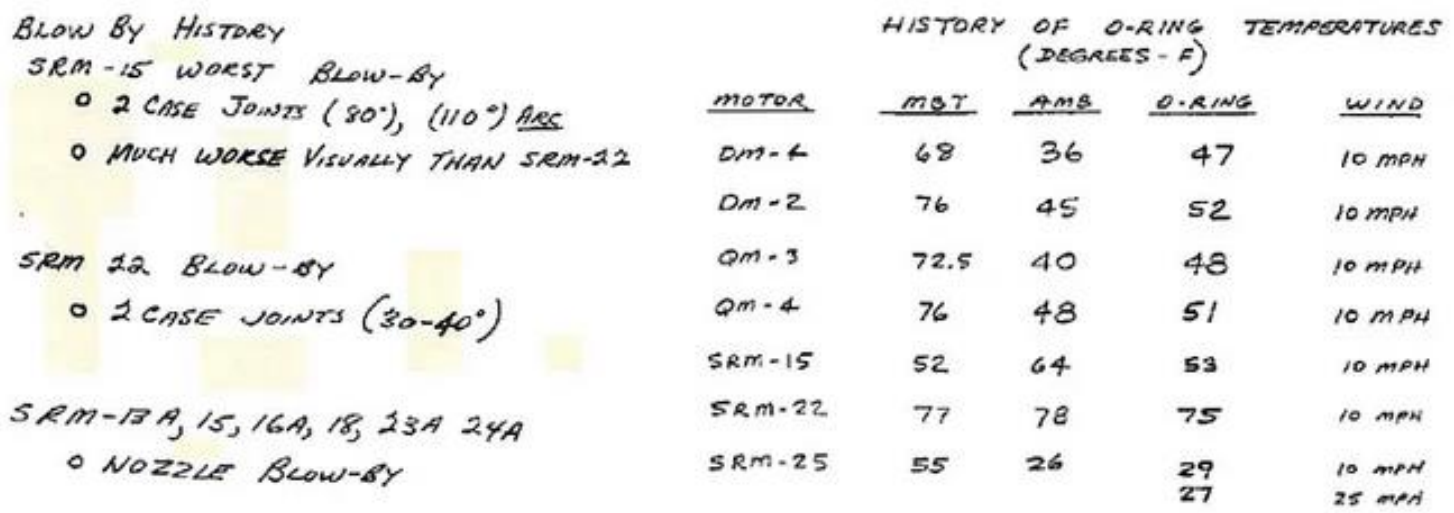
- Hal Varian, the chief economist for Google, calls Statistics and Data Science the new sexy profession.

Predictive Analytics



In-Match Win Probabilities for World Cup 2018: Data Source: 538

NASA: To Launch or Not to Launch?



BLOW BY HISTORY

SRM-15 WORST BLOW-BY

- 2 CASE JOINTS (80°), (110°) ARC
- MUCH WORSE VISUALLY THAN SRM-22

SRM 22 BLOW-BY

- 2 CASE JOINTS (30-40°)

SRM-13A, 15, 16A, 18, 23A 24A

- NOZZLE BLOW-BY

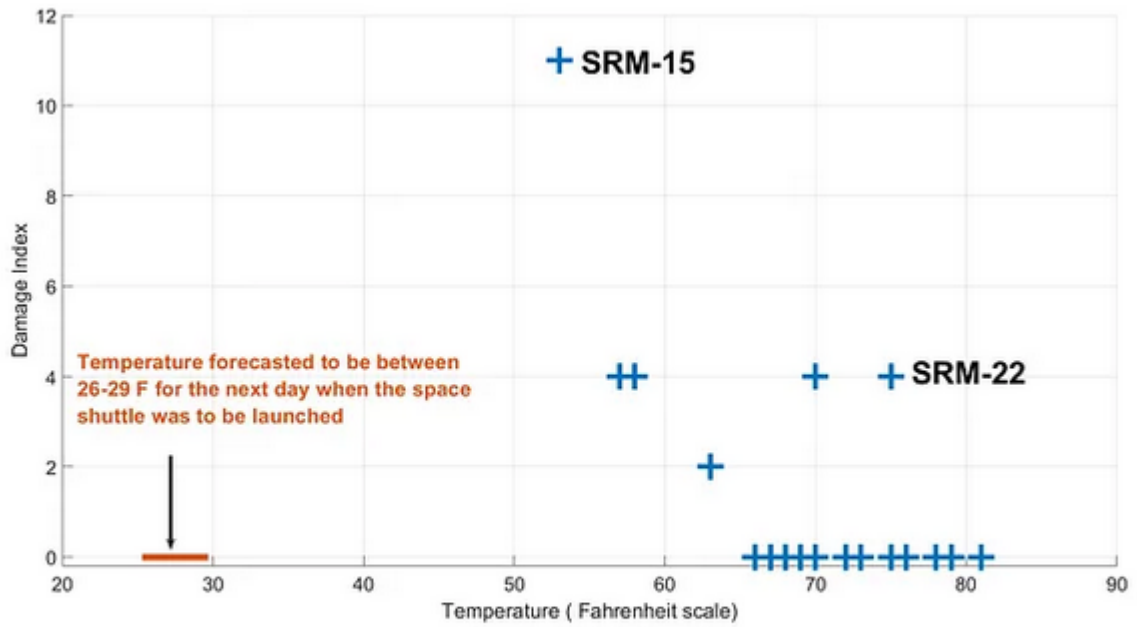
HISTORY OF O-RING TEMPERATURES (DEGREES - F)

MOTOR	MBT	AMB	O-RING	WIND
DM-4	68	36	47	10 MPH
DM-2	76	45	52	10 MPH
QM-3	72.5	40	48	10 MPH
QM-4	76	48	51	10 MPH
SRM-15	52	64	53	10 MPH
SRM-22	77	78	75	10 MPH
SRM-25	55	26	29 27	10 MPH 25 MPH

2 of the 13 charts presented by the engineers from Morton Thiokol, from [1]

https://medium.com/codex/5-thought-provoking-data-science-lessons-from-the-tragic-nasa-challenger-crash-that-killed-7-8a17043dc43b

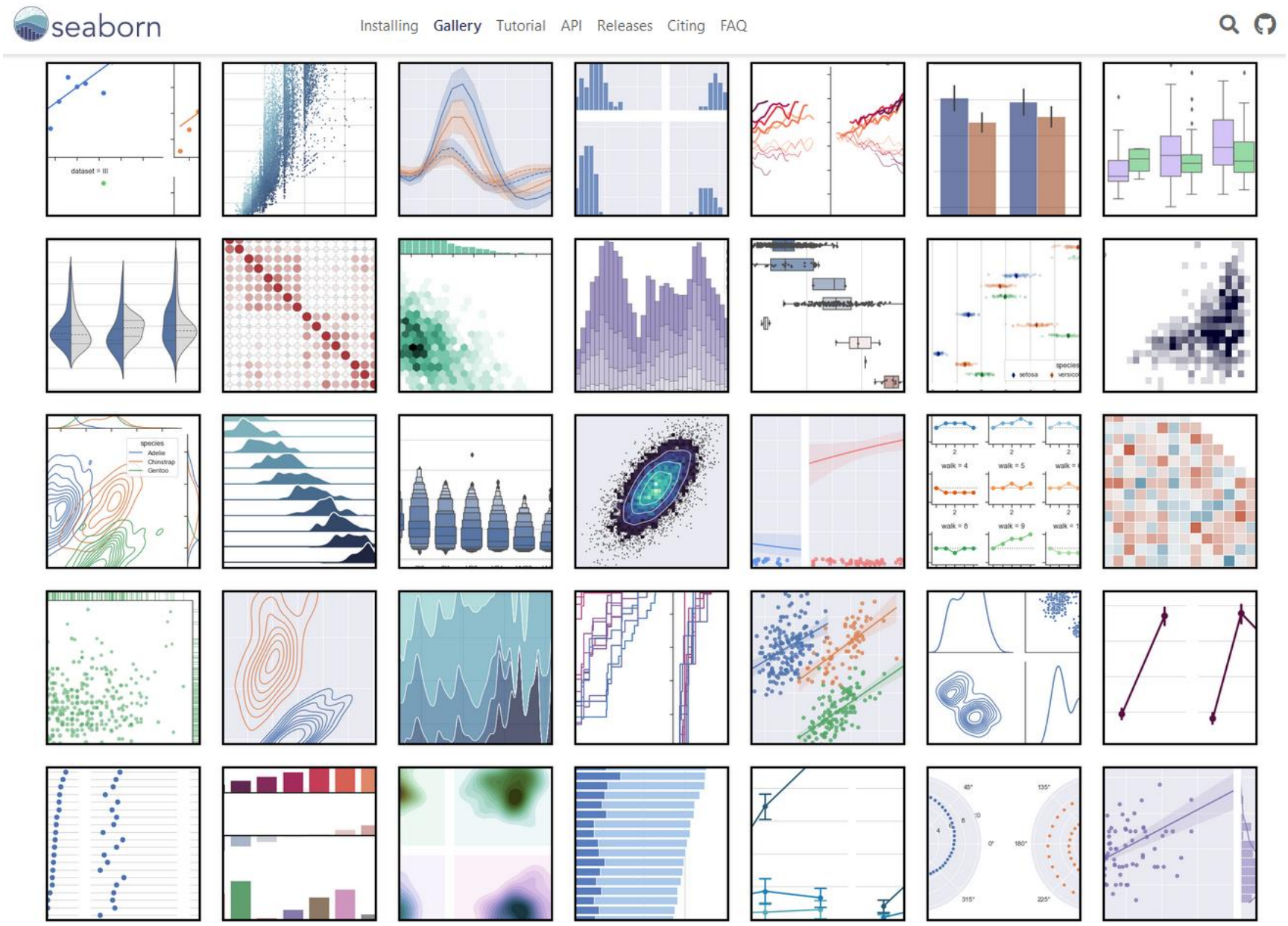
NASA: To Launch or Not to Launch?



Scatter Plot showing the damage Index of the O-rings against temperature recreated using data from [1], which now also shows the forecasted temperature for the next day which is literally off the charts! (Image by Author)

https://medium.com/codex/5-thought-provoking-data-science-lessons-from-the-tragic-nasa-challenger-crash-that-killed-7-8a17043dc43b

Data Presentation



Target

- Store wanted to identify pregnant women based on the products that they purchased.
- All major companies now have a *Predictive Analytics* group.
- Target's group built a model to predict with high accuracy whether a woman was pregnant and when the due date was. They then sent baby advertisements.
- In some ways the model worked too well!

Ways that Big Data/Data Science is transforming the (business) world

Education
- UC Berkeley's Fastest-Growing Class Is Data Science 101
- Tedx Talk

Fashion
- Stitch Fix
- Hush, which is a makeup app:

Retail Stores and Marketing
- Target
- Social Media Marketing
- Satellite tracking

Data Science for Social Good... e.g., using data science to improve traffic safety and other social goods:
- U Chicago Projects
- U Washington Projects

Data Privacy and the Internet
- Consumer Data Privacy
- A/B Testing

Health Care
- How Big Data is Changing Health Care

Transportation
- Uber

Law
- How Big Data is Disrupting the Legal Profession

Human Resources
- Big Data in Human Resources

Accounting
- Merging accounting with 'big data' science
- Data Science in Auditing

Investment banking
- An investment management company that puts algorithms and data at its center:

International development / Sustainable development
- Big data and sustainable development

Nonprofit, fundraising, development
- How nonprofits use big data to change the world
- Turning Your Nonprofit's Data Into Meaningful Information
- A Fundraiser's Secret Weapon: Data Analytics

Real estate
- Big Data Applications in Real Estate Analysis
- How Big Data is Disrupting Commercial Real Estate

Analytics Software
- APT

Compliance
- Data Analytics and Compliance

FORBES > TECH

How Target Figured Out A Teen Girl Was Pregnant Before Her Father Did

Kashmir Hill Former Staff
Welcome to The Not-So Private Parts where technology & privacy collide

Feb 16, 2012, 11:02am EST

Pole identified 25 products that when purchased together indicate a women is likely pregnant. The value of this information was that Target could send coupons to the pregnant woman at an expensive and habit-forming period of her life.

"My daughter got this in the mail!" he said. "She's still in high school, and you're sending her coupons for baby clothes and cribs? Are you trying to encourage her to get pregnant?"

The manager didn't have any idea what the man was talking about. He looked at the mailer. Sure enough, it was addressed to the man's daughter and contained advertisements for maternity clothing, nursery furniture and pictures of smiling infants. The manager apologized and then called a few days later to apologize again.

On the phone, though, the father was somewhat abashed. "I had a talk with my daughter," he said. "It turns out there's been some activities in my house I haven't been completely aware of. She's due in August. I owe you an apology."

Overview of Python

Python was originally developed by Guido van Rossum over the Christmas holiday break in 1989.
- named the language after the British comedy group Monty Python, and many early documentation examples and code snippets made reference to things found in Monty Python skits and movies.

Version 1.0 was released in 1994
Python version 2.0 (released in 2000) and version 3.0 (released in 2008) contained many features and enhancements

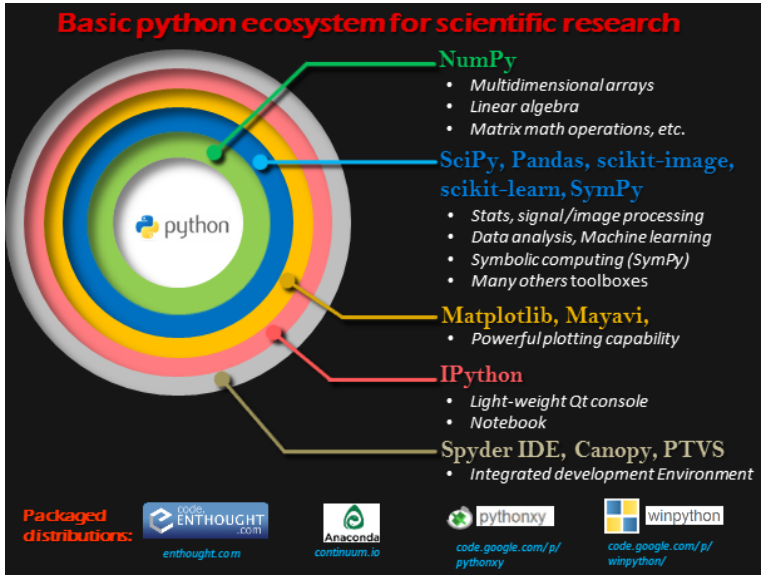
Python's rich ecosystem of tools and packages for numerical and scientific computing, graphics and data visualization, database access, statistics, and machine learning make it crucial for data science

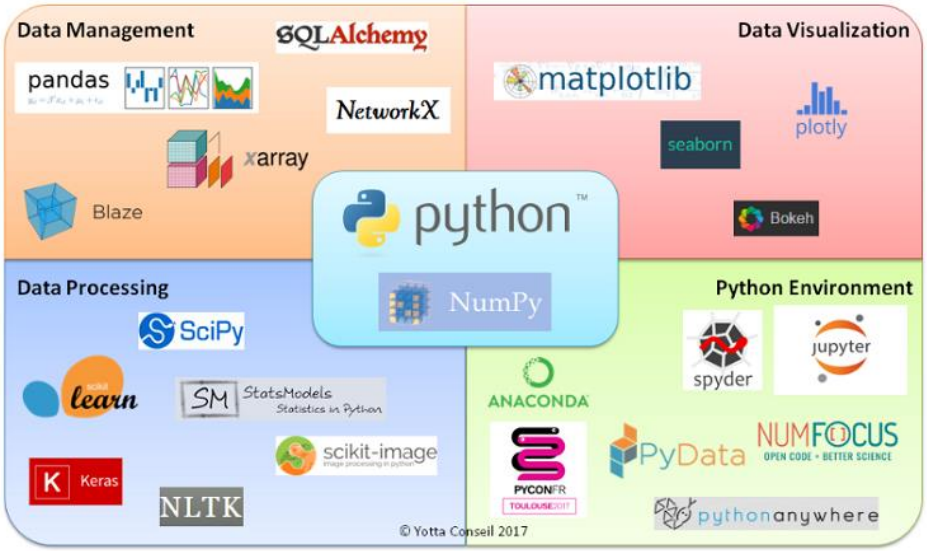
Overview of Python

Python is an elegant programming language with a lot of built-in useful tools, but it's also an ecosystem.

It is a general-purpose programming language, meaning that it wasn't developed to support some particular area.

It's a comprehensive set of packages, tools, and utilities, and developers who use Python, that have really added a lot of value for people





Python is an interpreted programming language: statements are processed one at a time by a program known as an interpreter. This provides a high level of interactivity, where users can explore and interrogate data or prototype different sorts of analyses or data visualizations.

There are multiple interpreters available for use with Python programs, such as the default python interpreter; the enhanced IPython, which is well-suited for interactive work; web-based Jupyter notebooks, which tie together code, analyses, documentation, and graphics; and integrated development environments (IDEs), which unite code editors, interpreter sessions, and data explorer tools.

Python's built-in functionality makes it an excellent tool to use immediately for carrying out numerical calculations.

Overview of Python

- IDE- Interactive Development Environment
  - Jupyter notebooks + Jupyter notebook extensions
  - Others (PyCharm, Visual Studio, Rstudio, etc)
- Basic Python Types
  - Numeric, sequence, text, Boolean, Dictionaries
- Control Flow
  - Conditional statements
  - For loop
  - List comprehension

Overview of Python

List comprehensions:

- Create lists from other lists, DataFrame columns, etc.
- Single line of code
- More efficient than using a for loop

Overview of Python

Python Method vs Functions

1.Method is called by its name, but it is **associated to an object** (dependent).
2.A method definition always includes **'self'** as its first parameter.
3.A method is **implicitly passed to the object** on which it is invoked.
4.It **may or may not return any data.**
5.A method **can operate on the data (instance variables) that is contained by the corresponding class**

Difference between Python Methods vs Functions

METHODS	FUNCTIONS
Methods definitions are always present inside a class.	We don't need a class to define a function.
Methods are associated with the objects of the class they belong to.	Functions are not associated with any object.
A method is called 'on' an object. We cannot invoke it just by its name	We can invoke a function just by its name.
Methods can operate on the data of the object they associate with	Functions operate on the data you pass to them as arguments.
Methods are dependent on the class they belong to.	Functions are independent entities in a program.
A method requires to have 'self' as its first argument.	Functions do not require any 'self' argument. They can have zero or more arguments.

Overview of Python

Python Method vs Functions

In [12]:
```
import pandas as pd
sat_df = pd.read_csv('chapter2_data.csv')
sat_df.head()

# The dataset represents results from a sales and product satisfaction survey (500 consumers)
# iProdSAT: product satisfaction
# iSalesSAT: experience with the sales person
# Segment: each respondent is assigned a numerically coded segment
# iProdREC: likelihood to recommend the product
# iSalesREC: likelihood to ecommend the same sales person
```

Out[12]:

	iProdSAT	iSalesSAT	Segment	iProdREC	iSalesREC
0	6	2	1	4	3
1	4	5	3	4	4
2	5	3	4	5	4
3	3	3	2	4	4
4	3	3	3	2	2

In [6]:
```
sat_df.describe()
```

Out[6]:

Overview of Python

RESERVED KEYWORDS

False	continue	from	not
None	def	global	or
True	del	if	pass
and	elif	import	raise
as	else	in	return
assert	except	is	try
break	finally	lambda	while
class	for	nonlocal	with
			yield

Overview of Python

Containers and Python's Built-in Types

Containers, also known as collections, are composite data types that (as the name suggests) contain other data types. They are useful for bundling together sets of related data rather than having separate variables for each individual data element.

You then iterate through the items in the list so that you can get everything you need.

Containers simplify both the process of storing sets of related data items and the process of operating on them.

Different container types bundle data elements in different ways and are intended for different purposes. Containers are common to almost all programming languages.

Overview of Python

Containers and Python's Built-in Types

In Python, there are several containers that are built in as part of the language and others that are accessible in additional libraries that you can access from within Python. Some of the key built-in Python containers are:

- **lists:** for storing a sequence of items in a specific order
- **dictionaries:** for associating unique members of one set of items (keys) with members of another set (values)
- **sets:** for storing a group of unique items in no specific order
- **strings:** for storing a sequence of textual information (e.g., to produce words, sentences, paragraphs)
- **tuples:** like lists, for storing a sequence of items in a specific order, but cannot be modified once created

Operating on data containers lies at the core of data science. A data scientist using Python needs to first understand how these core built-in container types operate since they are both useful in their own right and form the basis of other, more complex containers provided by external libraries.

Overview of Python

Python lists:

Group objects in specific order

Useful when ordering of objects is important

Can contain different types of Python objects (including other lists)

List items:

Are ordered sequentially and indexed by position in list (starting at 0)

Can be accessed by their position

Can be reassigned by their position (lists are mutable)

Lists can be sliced and concatenated

Python strings:

Are ordered groups of textual "characters"

Can contain letters, numbers, whitespace, unicode characters, etc.

Can create with matched pairs of single, double, or triple quotes (allows for embedded strings)

Can be indexed and sliced like lists

Are immutable like tuples

Can be concatenated with the + operator

Overview of Python

Python dictionaries:

Group Python objects by mapping keys (k) to values (v)

Associate one set of data with another (k-v pairs)

Are created with curly brackets

Are similar to lists but identified by user-specified keys rather than an integer index indicating position

Key-value pairs appear in no guaranteed order

Within dictionaries:

Entries are accessed via square brackets

An item can be accessed by its key

An item can be reassigned by its key (dictionaries are mutable)

Additionally:

Many Python objects are valid as keys, but not mutable containers like lists and dictionaries

Dictionaries cannot be sliced like lists (no order)

Overview of Python

Python sets:

A group of unique Python objects with no intrinsic order

Lookup is very fast compared to list

Are like mathematical sets

Overview of Python

Populate a list with a for loop

```
nums = [12, 8, 21, 3, 16]
new_nums = []
for num in nums:
    new_nums.append(num + 1)
print(new_nums)
```

```
[13, 9, 22, 4, 17]
```

A list comprehension

```
nums = [12, 8, 21, 3, 16]
new_nums = [num + 1 for num in nums]
print(new_nums)
```

```
[13, 9, 22, 4, 17]
```