Use Machine Learning to Find Your Next Job

Samuel Taylor

Job recommendations for 2017-09-03

assistant@samueltaylor.org Sep 3

to sgt

Sr. Machine Learning / Artificial Intelligence Engineer @ ClosedLoop.ai - http://www.indeed.com/cmp/ClosedLoop/jobs/Senior-Machine-Learning-f3f3a19d0d75b818

Data Engineer @ Austin Fraser - https://www.austinfraser.com/en-us/job/bbbh8350-data-engineer-1503529772/?utm_source=Indeed&utm_medium=organic&utm_campaign=Indeed

AppSumo - Python developer @ AppSumo - https://boards.greenhouse.io/appsumocareers/jobs/738433?gh_src=doqnew1

Back-End Developer (Python) @ Beyond - https://boards.greenhouse.io/beyond/jobs/814873?gh_src=ebmk7v1

Senior Back-End Developer @ Beyond - https://boards.greenhouse.io/beyond/jobs/814896?gh_src=1xoahl1

Software Development Principal Engineer - Austin, TX @ Dell - https://dell.taleo.net/careersection/2/jobdetail.ftl?job=17000FQB&tz=GMT-05:00&src=JB-11346

Outline

- Introduction
- Asking the right question
- Gathering data
- Analysis
- Deploying

Outline

- **Introduction**
- Asking the right question
- Gathering data
- Analysis
- Deploying

Outline

- Introduction
- **Asking the right question**
- Gathering data
- Analysis
- Deploying

1. Have a problem

Machine learning?

Machine learning

- Goal: Find *f(x)*
- Problem: *f(x)* is unknown
- But: we can measure some points from *f(x)*
- Algorithms to find a *g(x)* that approximates *f(x)*

Supervised

- Regression
- Classification

Unsupervised

- Clustering

Other stuff

- Reinforcement

Supervised

- Regression
- Classification

Unsupervised

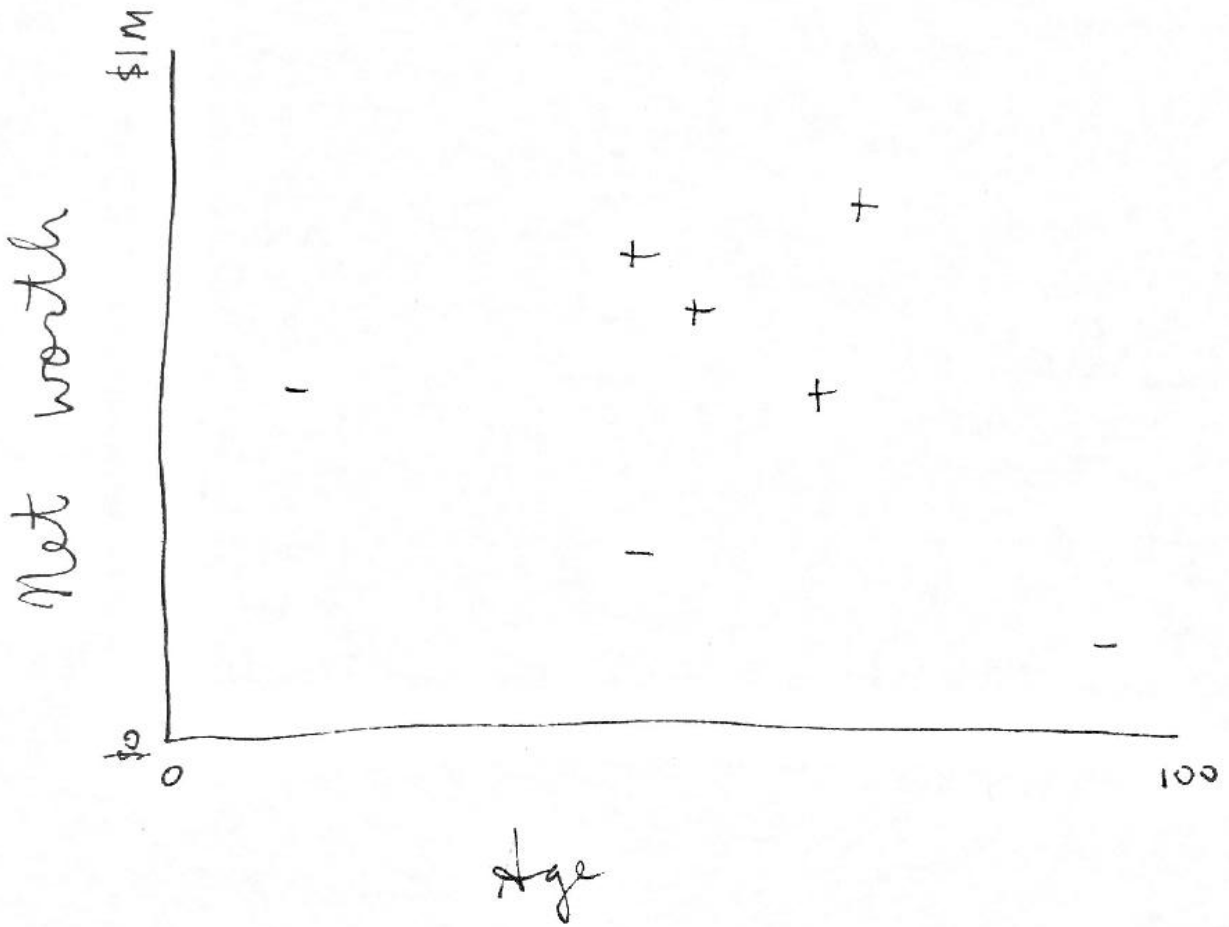
- Clustering

Other stuff

- Reinforcement

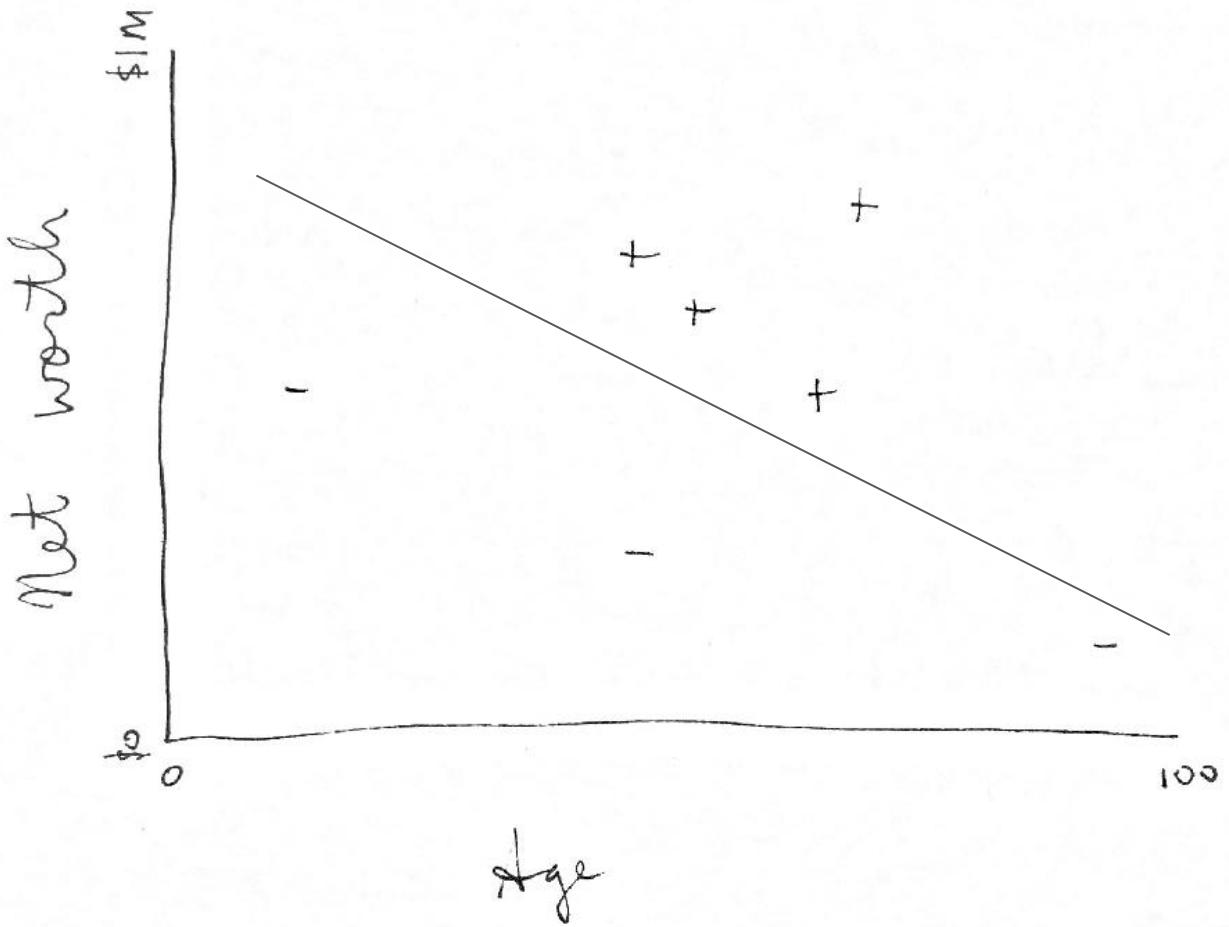
Supervised

- Classification



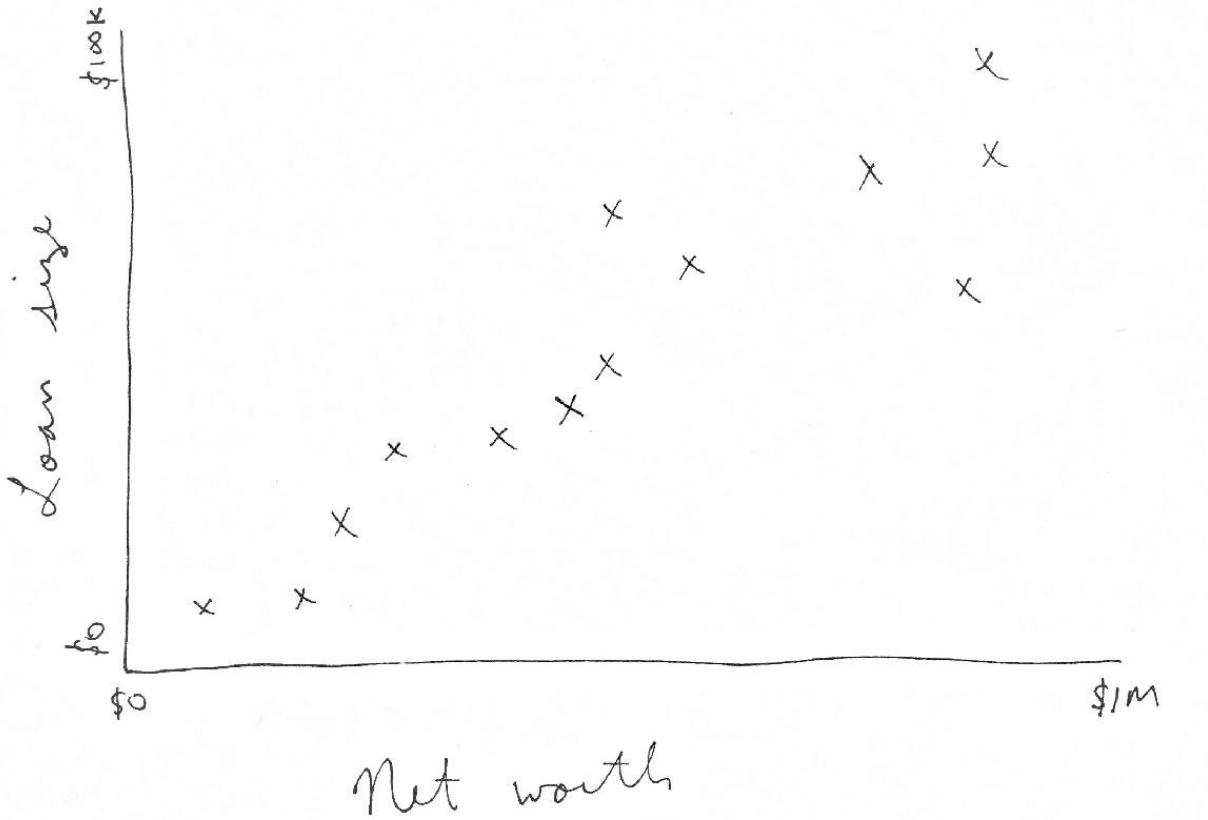
Supervised

- Classification



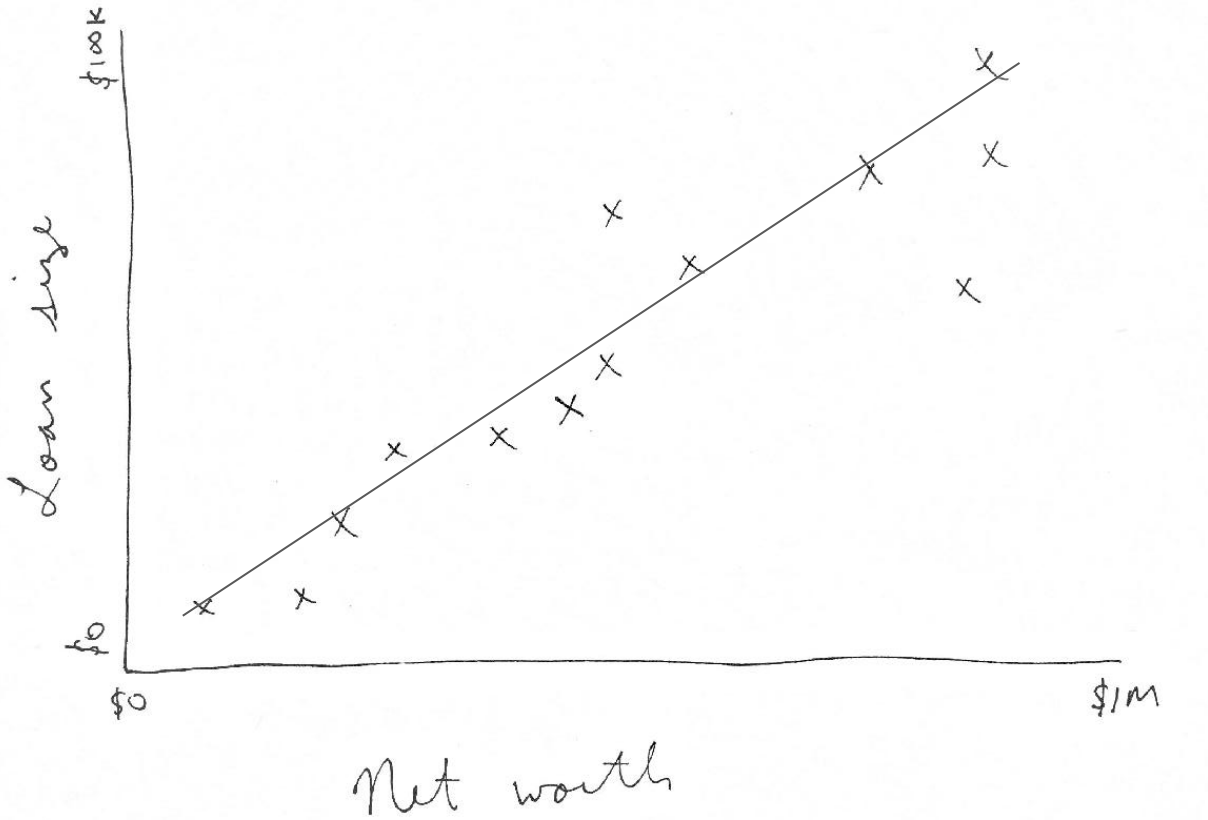
Supervised

- Classification
- Regression



Supervised

- Classification
- Regression



2. Phrase the question

Outline

- Introduction
- Asking the right question
- **Gathering data**
- Analysis
- Deploying

Existing

- Google it
- Government
- data.world

Existing

- Google it
- Government
- data.world

Create it

- Spreadsheet
- IFTTT
- Web scraping

Protip: obey robots.txt

	A	B	C	D	E
1	**Title**	**Company**	**U**	**Link**	**Sounds cool**
2	Principal Software Architect - Austin	General Electric	/r	Link	1
3	ASIC Power Estimation Developer (Excel-I	Encore Semi	/r	Link	0
4	Memory Subsystem Verification Engineer	Encore Semi	/r	Link	0
5	Senior DevOps Engineer	KIBO Software	/r	Link	0
6	Senior Manager of Software Engineering	MaxPoint	/r	Link	1
7	Data Analyst	Amherst	/r	Link	0
8	Senior Data Engineer	Visa	/r	Link	1
9	Product Development Engineer	Advanced Micro Devices, Inc.	/r	Link	0
10	Systems Analyst	Visa	/r	Link	0
11	Lead Architect - Big Data	Farmers Edge	/r	Link	1
12	Object Storage Software Engineer	IBM	/r	Link	0
13	Principal Site Reliability Engineer	Pearson	/r	Link	0
14	Senior Software Development Engineer - S	Amazon Corporate LLC	/r	Link	0
15	Systems Administrator I	University of Texas at Austin	/r	Link	0
16	Senior Database Administrator	Acxiom	/r	Link	0
17	IT Support Representative	Becker Wright Consultants	/c	Link	0
18	Software Development Engineer - Silicon (	Amazon Corporate LLC	/r	Link	0
19	Software Developer	IBM	/r	Link	0
20	Sr. Product Development Engineer	Advanced Micro Devices, Inc.	/r	Link	0
21	Front end developer	IBM	/r	Link	0
22	Full Stack Software Engineer	Indeed	/r	Link	1

	A	B	C	D	E
1	Title	Company	U	Link	Sounds cool
2	Principal Software Architect - Austin	General Electric	/r	Link	1
3	ASIC Power Estimation Developer (Excel-I	Encore Semi	/r	Link	0
4	Memory Subsystem Verification Engineer	Encore Semi	/r	Link	0
5	Senior DevOps Engineer	KIBO Software	/r	Link	0
6	Senior Manager of Software Engineering	MaxPoint	/r	Link	1
7	Data Analyst	Amherst	/r	Link	0
8	Senior Data Engineer	Visa	/r	Link	1
9	Product Development Engineer	Advanced Micro Devices, Inc.	/r	Link	0
10	Systems Analyst	Visa	/r	Link	0
11	Lead Architect - Big Data	Farmers Edge	/r	Link	1
12	Object Storage Software Engineer	IBM	/r	Link	0
13	Principal Site Reliability Engineer	Pearson	/r	Link	0
14	Senior Software Development Engineer - S	Amazon Corporate LLC	/r	Link	0
15	Systems Administrator I	University of Texas at Austin	/r	Link	0
16	Senior Database Administrator	Acxiom	/r	Link	0
17	IT Support Representative	Becker Wright Consultants	/c	Link	0
18	Software Development Engineer - Silicon (	Amazon Corporate LLC	/r	Link	0
19	Software Developer	IBM	/r	Link	0
20	Sr. Product Development Engineer	Advanced Micro Devices, Inc.	/r	Link	0
21	Front end developer	IBM	/r	Link	0
22	Full Stack Software Engineer	Indeed	/r	Link	1

Existing

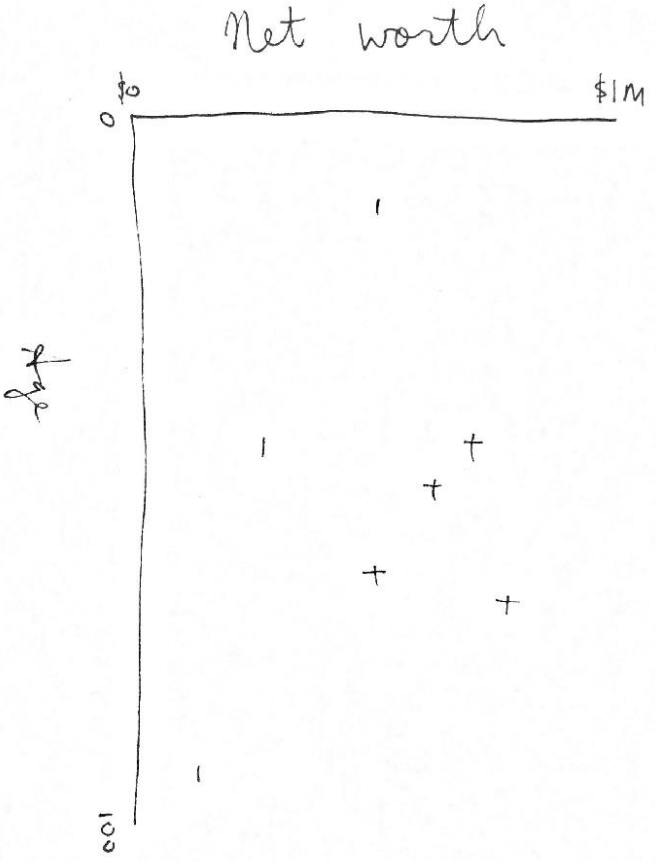
- Google it
- Government
- data.world

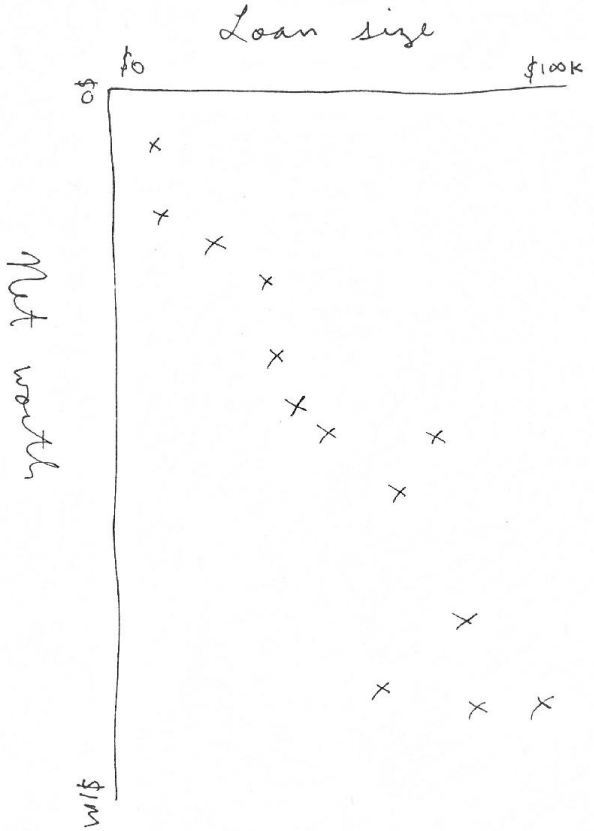
Create it

- Spreadsheet
- IFTTT
- Web scraping

Clean it

- Pandas
- scikit-learn


Net worth
$0
$1M
0
Age
100


Loan size
$0
$100K
$0
Net worth
$1M

(Sr. Data Engineer, sounds_cool=True)  (5, 1)

?

	Engi-neer	web	Applica-tions	sr	jr	analytics	software	data	developer
Sr. Web Applications Developer - Data Analytics	0	1	1	1	0	1	0	1	1
Jr. Software Developer	0	0	0	0	1	0	1	0	1
Sr. Data Engineer	1	0	0	1	0	0	0	1	0

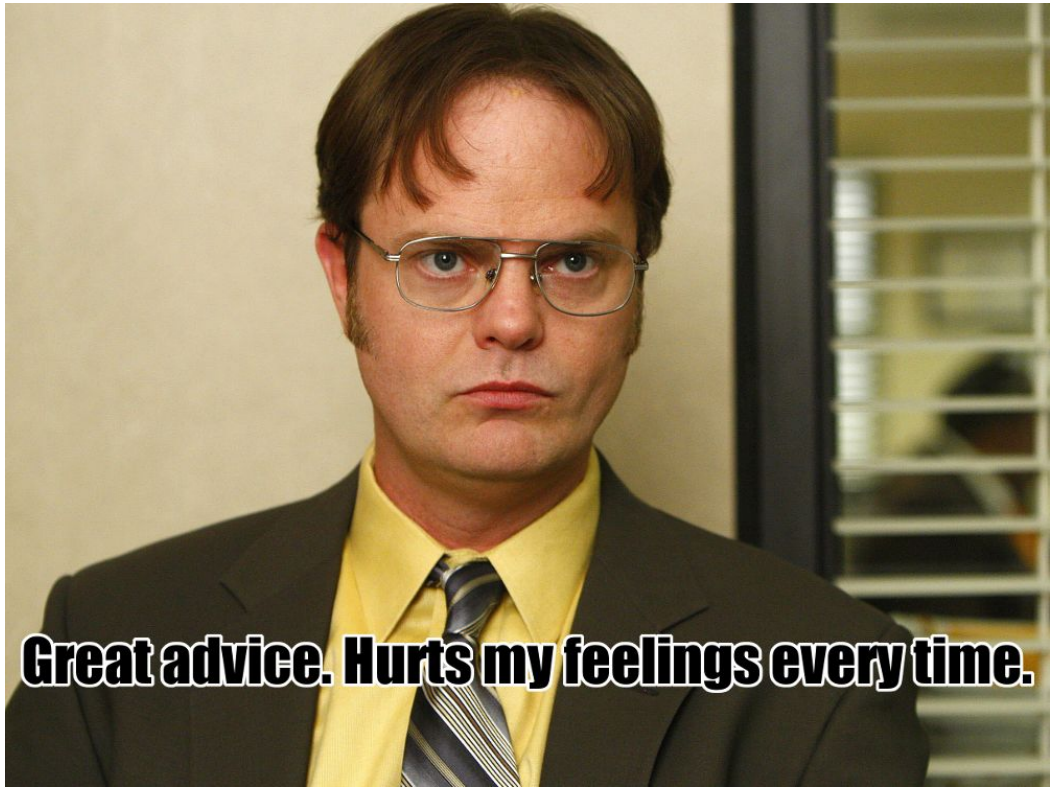
(Sr. Data Engineer, sounds_cool=True)

(1, 0, 0, 1, 0, 0, 0, 1, 0, 1)

Outline

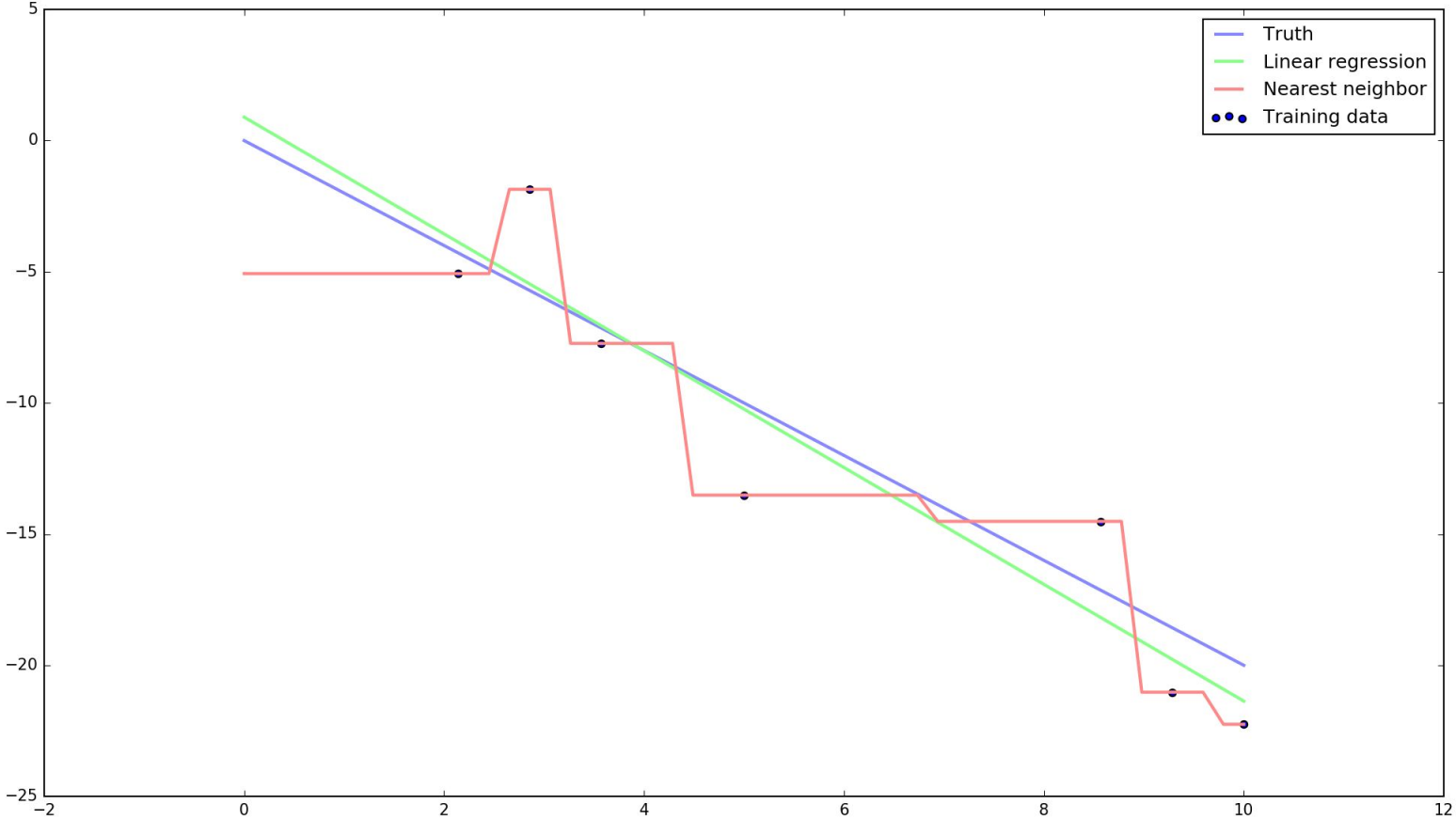
- Introduction
- Asking the right question
- Gathering data
- **Analysis**
- Deploying

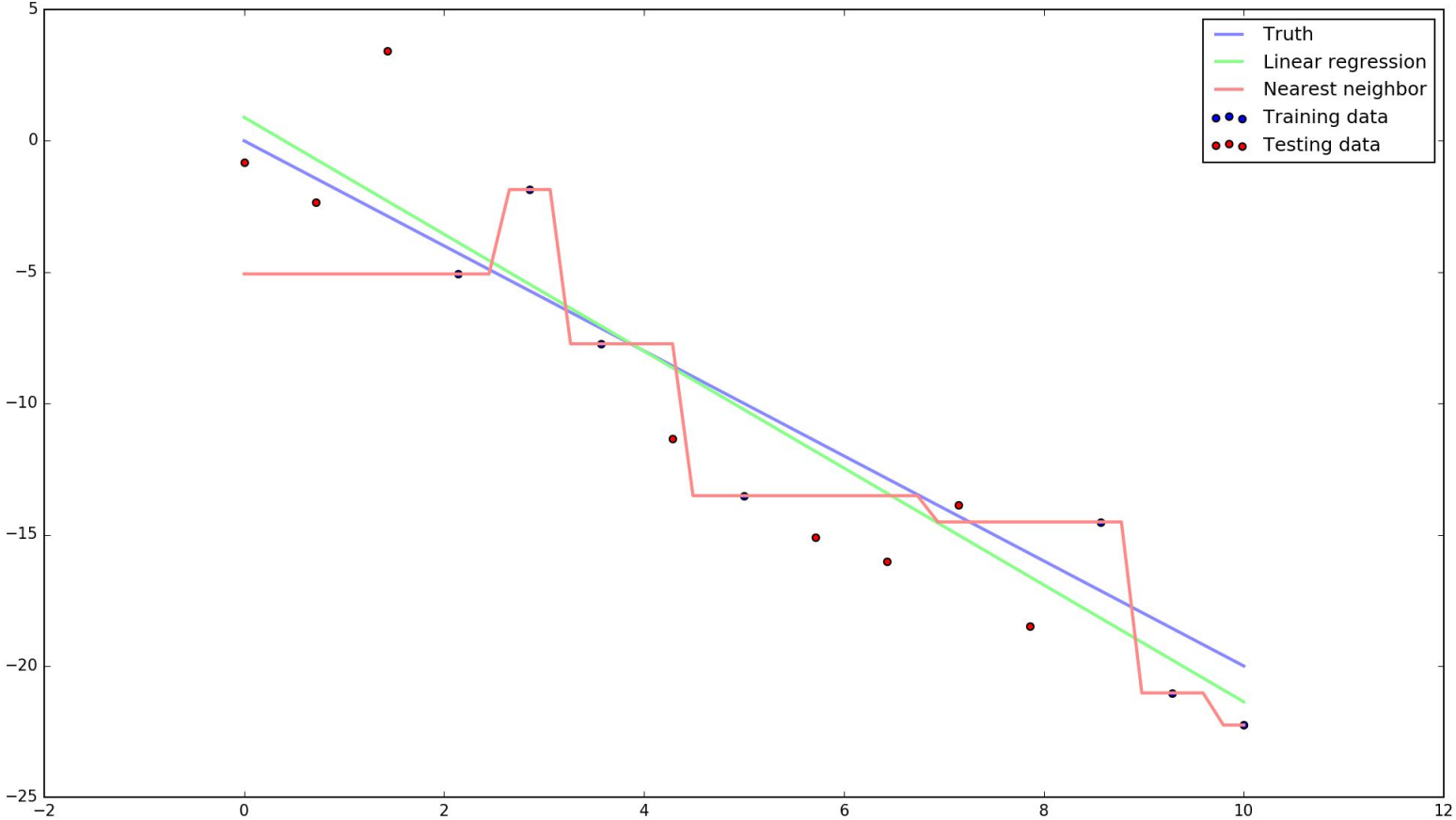
3. KISS



Theory

- Approximation-generalization tradeoff


Truth
Linear regression
Nearest neighbor
Training data
5
0
−5
−10
−15
−20
−25
−2
0
2
4
6
8
10
12


Truth
Linear regression
Nearest neighbor
Training data
Testing data
5
0
−5
−10
−15
−20
−25
−2
0
2
4
6
8
10
12

Theory

- Approximation-generalization tradeoff
- It’s just easier

Theory

- Approximation-generalization tradeoff
- It’s just easier

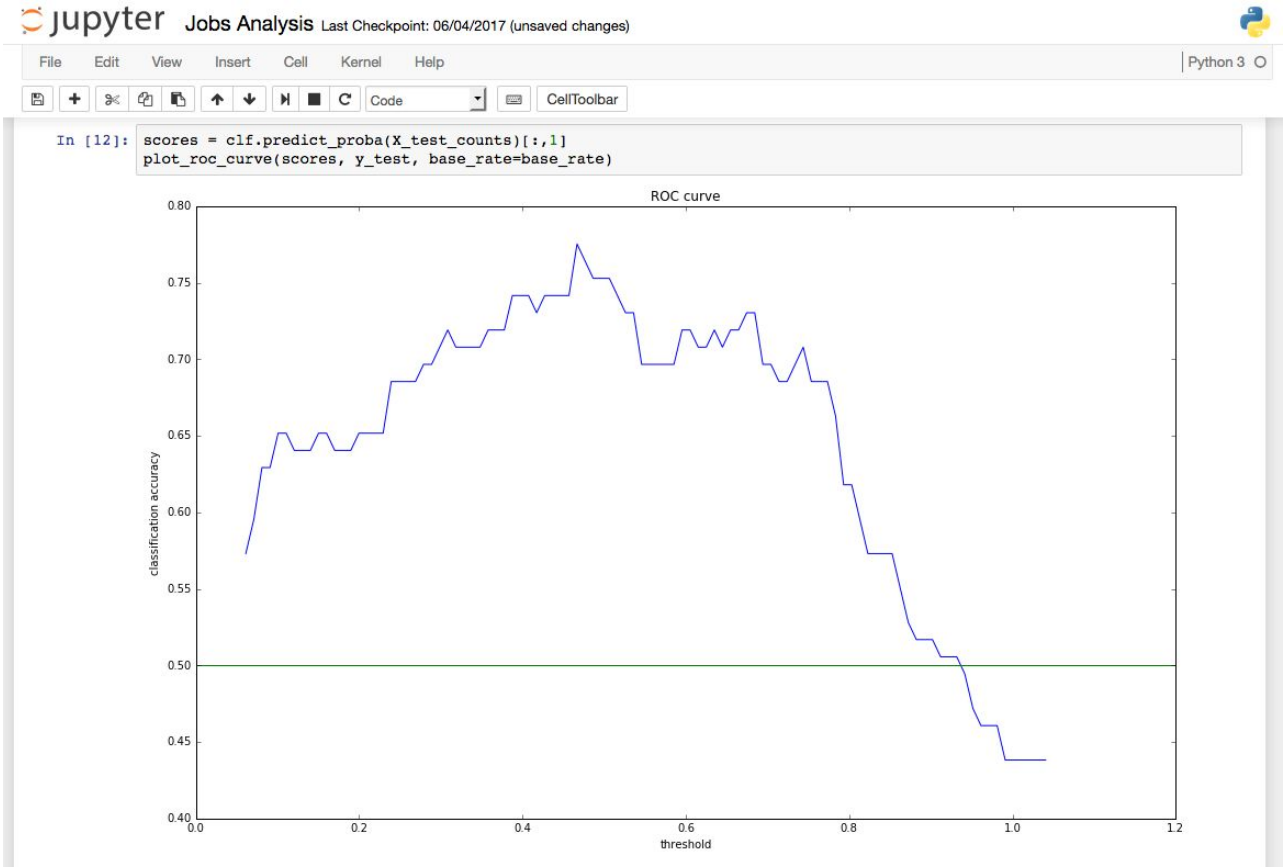
Practice

- Start with simple models
  - Linear regression
  - Logistic regression

```
X = rated_jobs['title'].as_matrix()
y = rated_jobs['sounds_cool'].as_matrix()

vect = CountVectorizer()
Xp = vect.fit_transform(X).toarray()
clf = LogisticRegression().fit(Xp, y)

new_job_ratings = clf.predict(new_jobs)

# array([ 0.,  0.,  0.,  1.,  0.,  0.,  0.,  1.,  0.,  0.])
```

Recommended tools

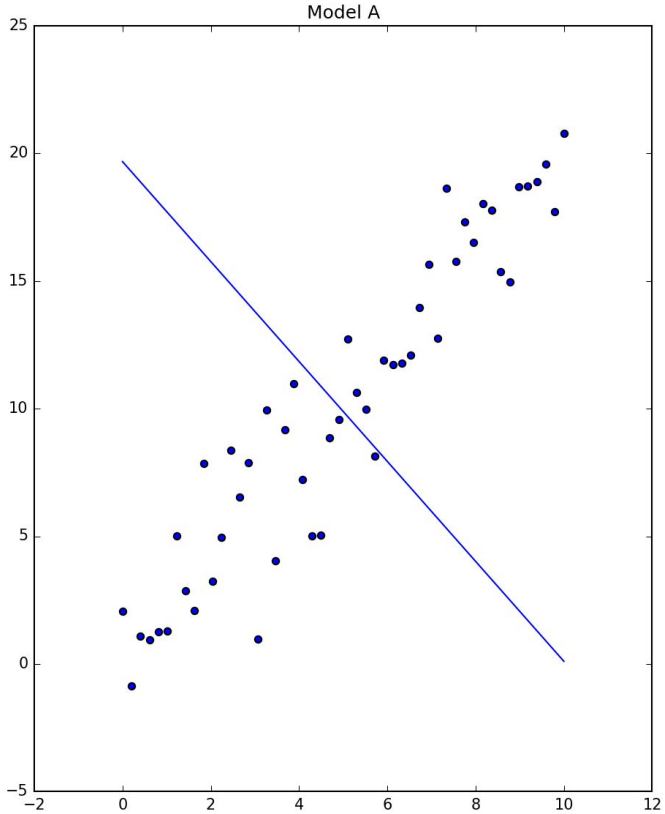
- Jupyter
- Pandas
- scikit-learn

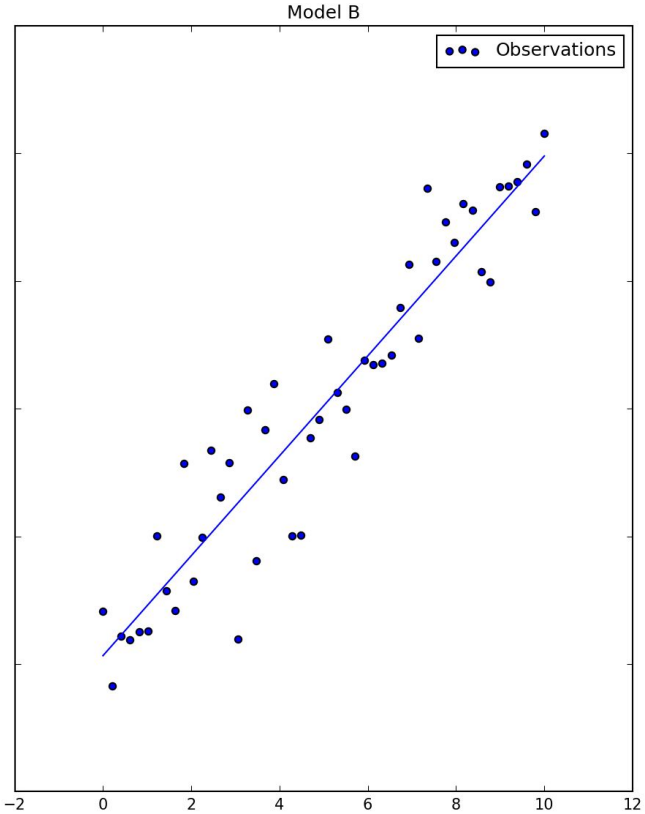


4. Test + iterate

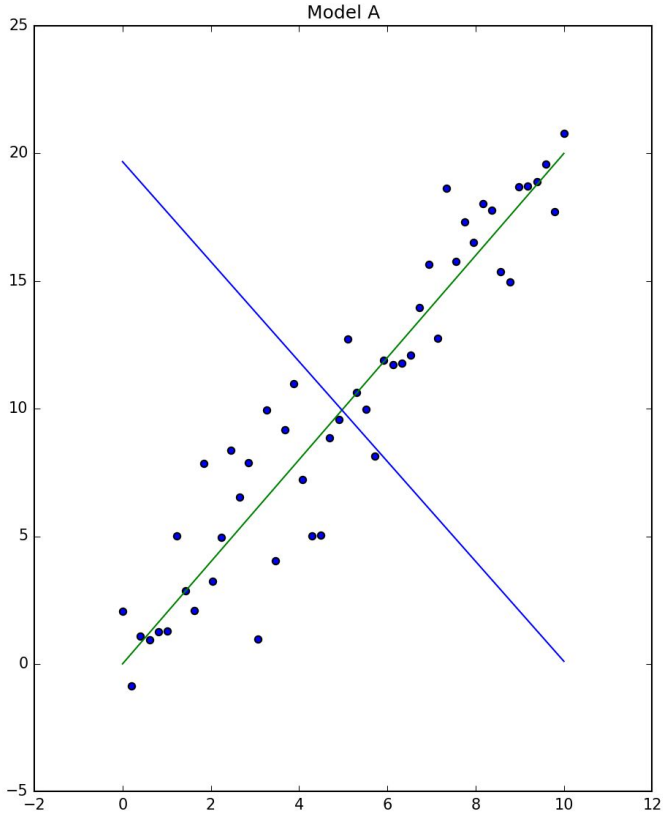
How accurate is it?

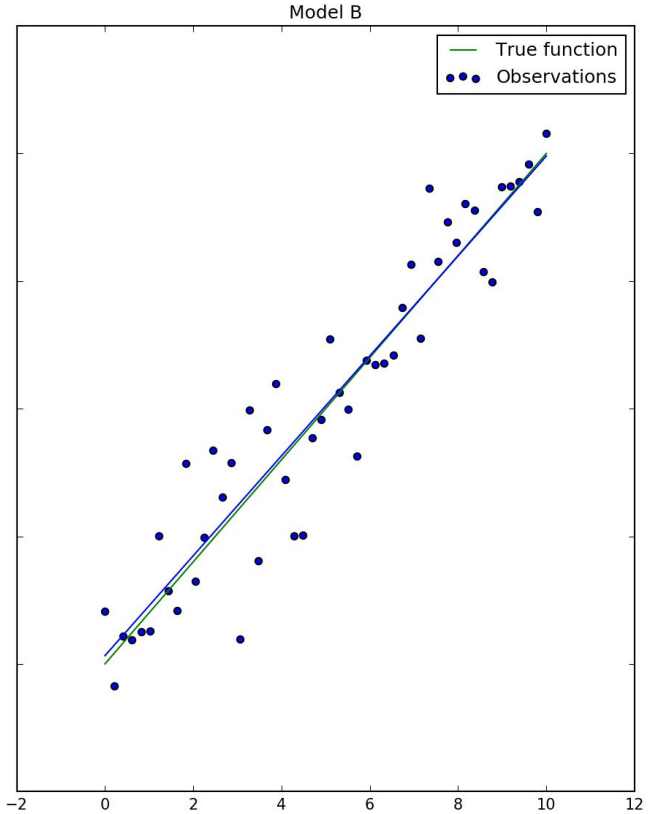
Measuring Error





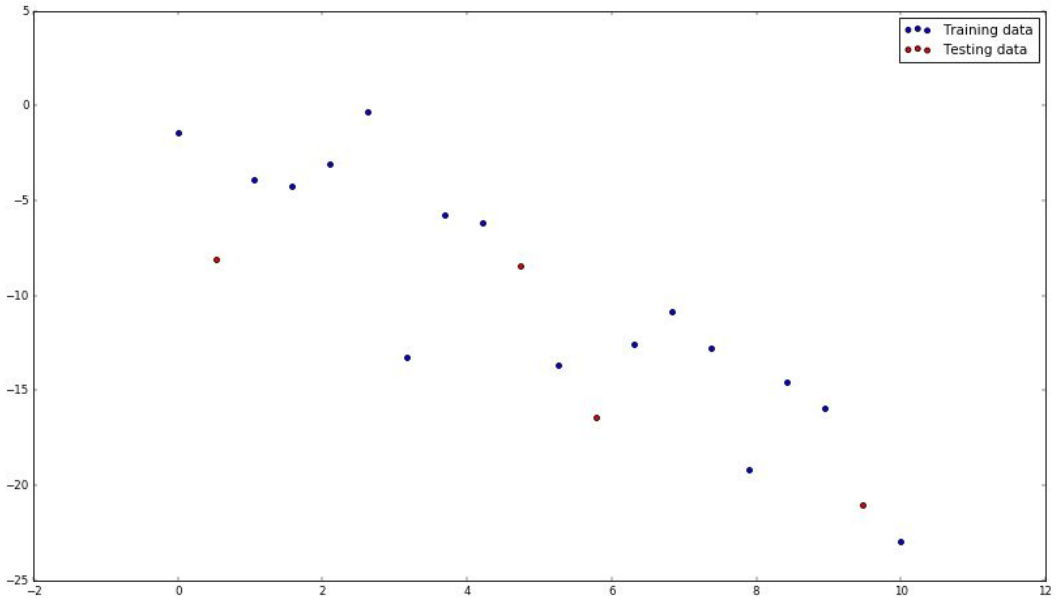
Measuring Error





Measuring Error

- Hold out some “testing data”

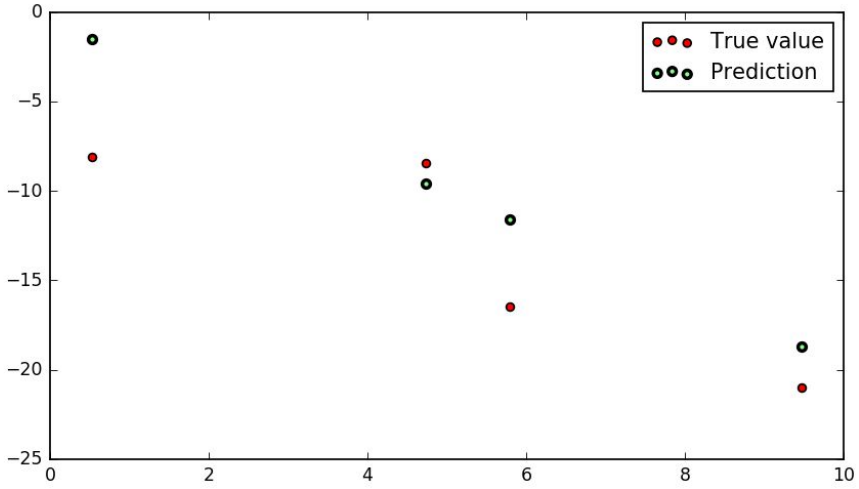


Measuring Error

- Hold out some “testing data”
- Compare test data to prediction
- Ideally: calculate the real cost of an error
  - Cost of false positive in nuclear warhead detection: **HIGH**
  - Cost of false positive in fingerprint recognition on my phone: **SIGNIFICANTLY LOWER**

Measuring Error

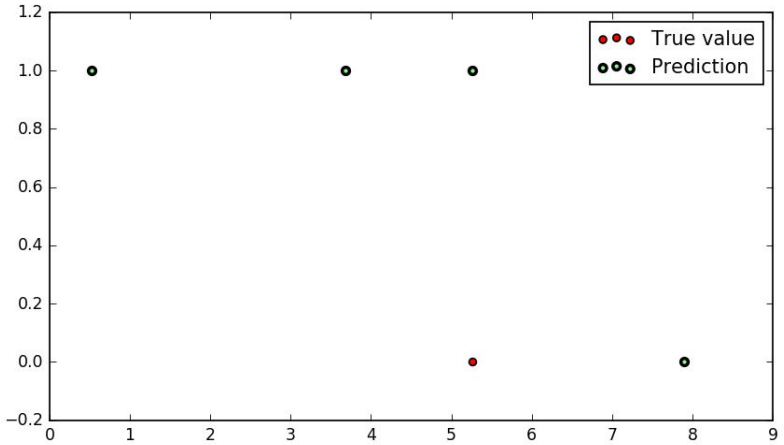
- Compare test data to prediction
- Common metric for regression: mean squared error
  - 18.35



Input	True	Predict	Diff	Sq. diff
0.53	-8.10	-1.51	-6.60	43.50
4.74	-8.47	-9.60	1.13	1.27
5.79	-16.45	-11.62	-4.83	23.30
9.47	-21.01	-18.70	-2.31	5.34

Measuring Error

- Compare test data to prediction
- Common metric for classification: mean classification error
  - 0.25



Input	True	Predict	Error?
0.53	1	1	0
3.68	1	1	0
5.26	0	1	1
7.89	0	0	0

Back to jobs!

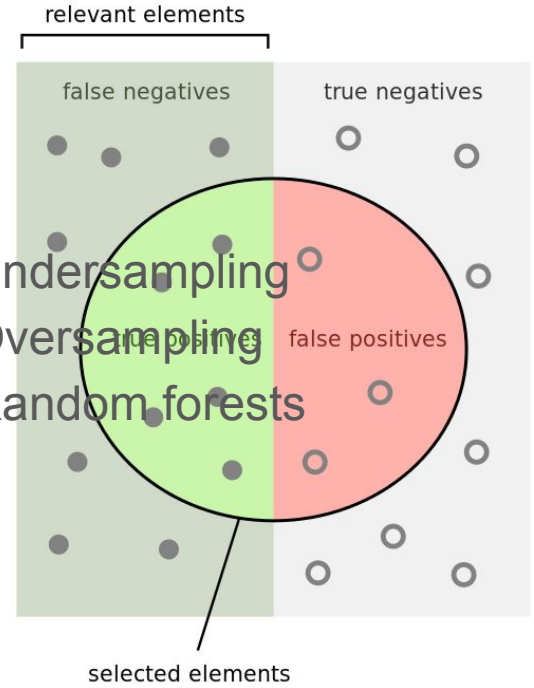
- Classification error: 0.197
  - Awesome!
- But wait, it’s just classifying everything as “not cool”
- Base rate for this problem is 0.197
  - No improvement

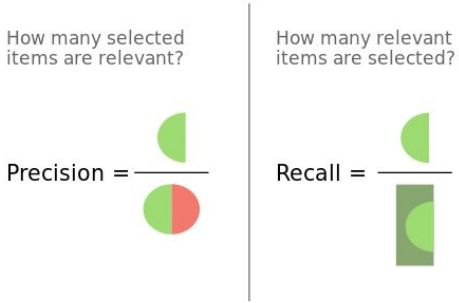
How skewed data makes me feel

Handling imbalanced classes

- Better error metrics
  - Precision
  - Recall

- Undersampling
- Oversampling
- Random forests





Walber

Outline

- Introduction
- Asking the right question
- Gathering data
- Analysis
- **Deploying**

```
X = rated_jobs['title'].as_matrix()
y = rated_jobs['sounds_cool'].as_matrix()

vect = CountVectorizer()
Xp = vect.fit_transform(X).toarray()
clf = LogisticRegression().fit(Xp, y)

new_job_ratings = clf.predict(new_jobs)

# array([ 0.,  0.,  0.,  1.,  0.,  0.,  0.,  1.,  0.,  0.])
```

Job recommendations for 2017-09-03

assistant@samueltaylor.org Sep 3

to sgt

Sr. Machine Learning / Artificial Intelligence Engineer @ ClosedLoop.ai - http://www.indeed.com/cmp/ClosedLoop/jobs/Senior-Machine-Learning-f3f3a19d0d75b818

Data Engineer @ Austin Fraser - https://www.austinfraser.com/en-us/job/bbbh8350-data-engineer-1503529772/?utm_source=Indeed&utm_medium=organic&utm_campaign=Indeed

AppSumo - Python developer @ AppSumo - https://boards.greenhouse.io/appsumocareers/jobs/738433?gh_src=doqnew1

Back-End Developer (Python) @ Beyond - https://boards.greenhouse.io/beyond/jobs/814873?gh_src=ebmk7v1

Senior Back-End Developer @ Beyond - https://boards.greenhouse.io/beyond/jobs/814896?gh_src=1xoahl1

Software Development Principal Engineer - Austin, TX @ Dell - https://dell.taleo.net/careersection/2/jobdetail.ftl?job=17000FQB&tz=GMT-05:00&src=JB-11346

4. Test + iterate

The Recruiter Repellant 3000

<demo>

4. Test + iterate
+ iterate
+ iterate

1. Have a problem
2. Phrase the question
3. Try the simplest thing
4. Test and iterate

More resources

- Learning from Data
- Practical Business Python


indeed®

Samuel Taylor

sgt@samueltaylor.org

@SamuelDataT