TECH BATTLE

GRAPHS VS MACHINE LEARNING

DENIS VRDOLJAK, BERKELEY DATA SCIENCE GROUP

HAIKAL PRIBADI, GRAKN AI

MACHINE LEARNING

VS.

GRAPHS

WHO WILL WIN??

THE CHALLENGE

HOW WELL CAN WE PREDICT A GOOD OR BAD CREDIT RATING FOR AN INDIVIDUAL?

THE CHALLENGE

CREDIT CARD FRAUD DATASET

- ORIGINAL DATASET PROVIDED BY PROFESSOR HANS HOFFMAN
- SERIES OF ATTRIBUTES INCLUDING AMOUNT IN CHECKING ACCOUNT, CREDIT HISTORY, AND CREDIT AMOUNT, AMONG OTHERS
- SOME ARE NUMERICAL, MOST ARE QUALITATIVE WITH 3 TO 6 OPTIONS

MACHINE LEARNING

FIRST A LITTLE CONTEXT

WHAT IS MACHINE LEARNING

- MACHINE LEARNING IS A SET OF TOOLS AND TECHNIQUES USED TO FIND PATTERNS, MAKE PREDICTIONS, AND IDENTIFY TRENDS IN LARGE DATASETS.

Machine learning

...a field of computer science that gives computers the ability to learn without being explicitly programmed.
(Wikipedia)

THE BLACK BOX PROBLEM

- COMMON ML ARCHITECTURES (LIKE NN'S) ARE A BLACK BOX
- THE UNDERLYING LOGIC IS HIDDEN FROM THE USER, AND THE PROGRAMMER
- THIS LIMITS THEIR EXPLANATORY USEFULNESS
- EXAMPLE: CREDIT SCORES

DEALING WITH THE BLACK BOX PROBLEM

- TECHNIQUES FOR UNDERSTANDING COMPLEX/BLACK-BOX ML MODELS:
- PCA
- MODEL REDUCTION
- PROBING
- ETC

MODEL REDUCTION EXAMPLE APPLICATION

- IN THE US, EVERY CREDIT DENIAL REQUIRES A LETTER SENT TO THE APPLICANT EXPLAINING THE REASON FOR THE DENIAL
- CREDIT APPLICATIONS ARE EVALUATED WITH NN BASED ML MODELS TO RATE CREDIT WORTHINESS
- A MODEL REDUCTION PROVIDES AN APPROXIMATION OF THE REASON FOR DENIAL
- THIS IS SENT TO THE CONSUMER TO COMPLY WITH FEDERAL LAW

MODEL REDUCTION EXAMPLE APPLICATION

Loan Decline Letter

Date

Name
Address
City, State, Zip

Dear {Name of applicant},

Thank you for your recent application. Your request for __a loan___ was carefully considered, and we regret that we are unable to approve your application at this time for the following reasons:

Your Income:
_____ is below our minimum requirement
_____ is insufficient to sustain payments on the amount of credit requested
_____ could not be verified

Your Employment:
_____ is not a sufficient length to qualify
_____ could not be verified

Your Credit History:
_____ of making payments on time was not satisfactory
_____ could not be verified

Your Application:
_____ lacks a sufficient number of credit references
_____ lacks acceptable types of credit references
_____ reveals that current obligations are excessive in relation to income

Other: _____

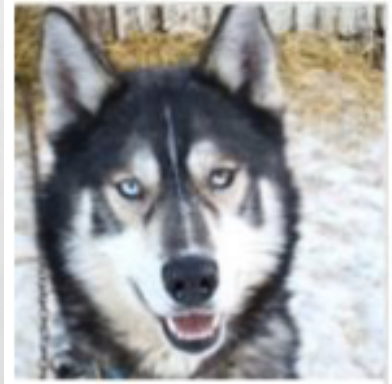
IS IT A WOLF? A BLACK BOX PROBLEM

- A HUSKY OR A WOLF

Predicted: wolf
True: wolf

Predicted: husky
True: husky

Predicted: wolf
True: wolf

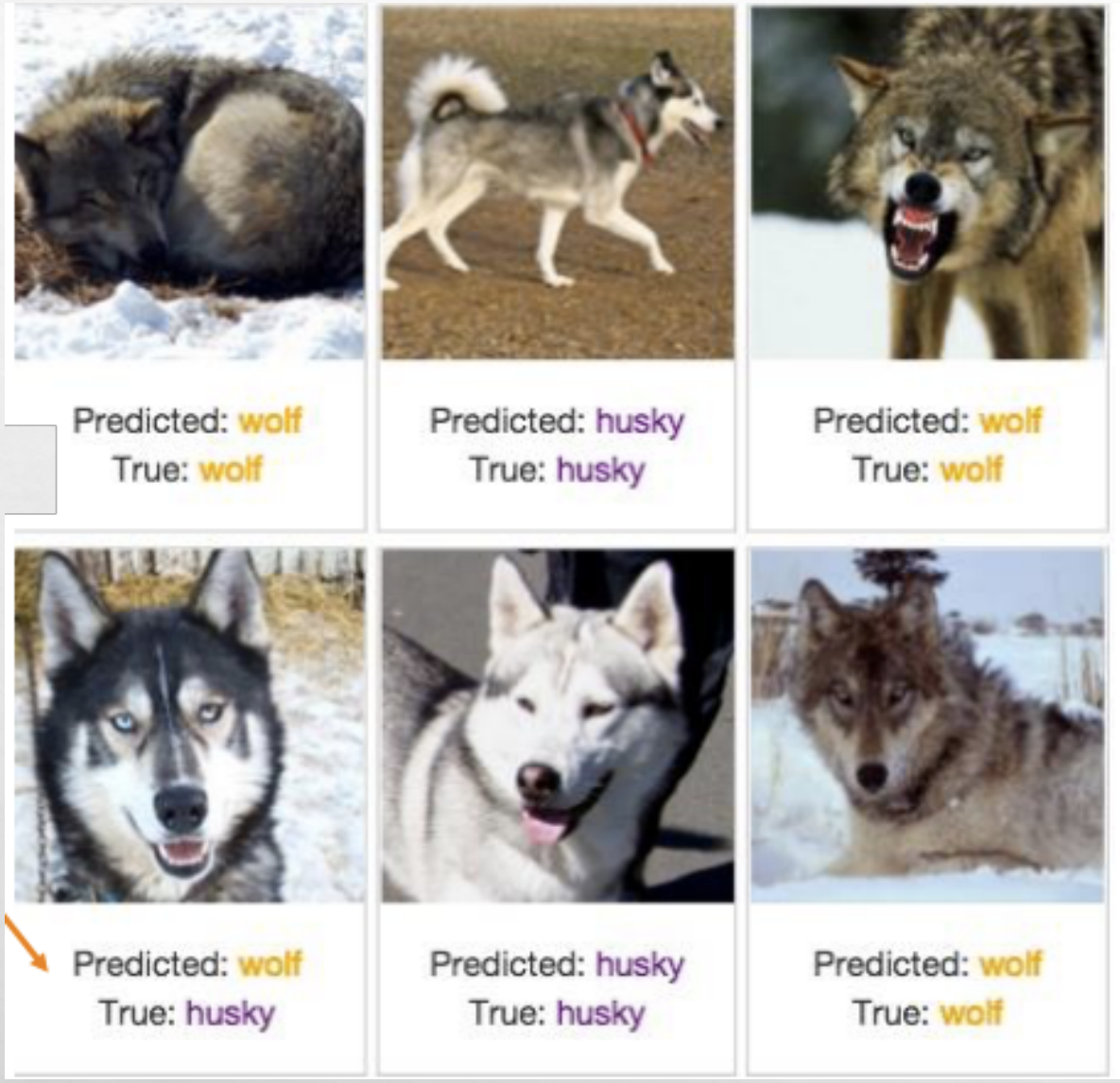


Predicted: wolf
True: husky

Predicted: husky
True: husky

Predicted: wolf
True: wolf

http://oc.acm.org/docs/7-12-2017%20-%20Sameer%20Singh%20-%20Explaining%20Black-Box%20ML%20Predictions.pdf



IT ENDS UP JUST BEING A SNOW DETECTOR!

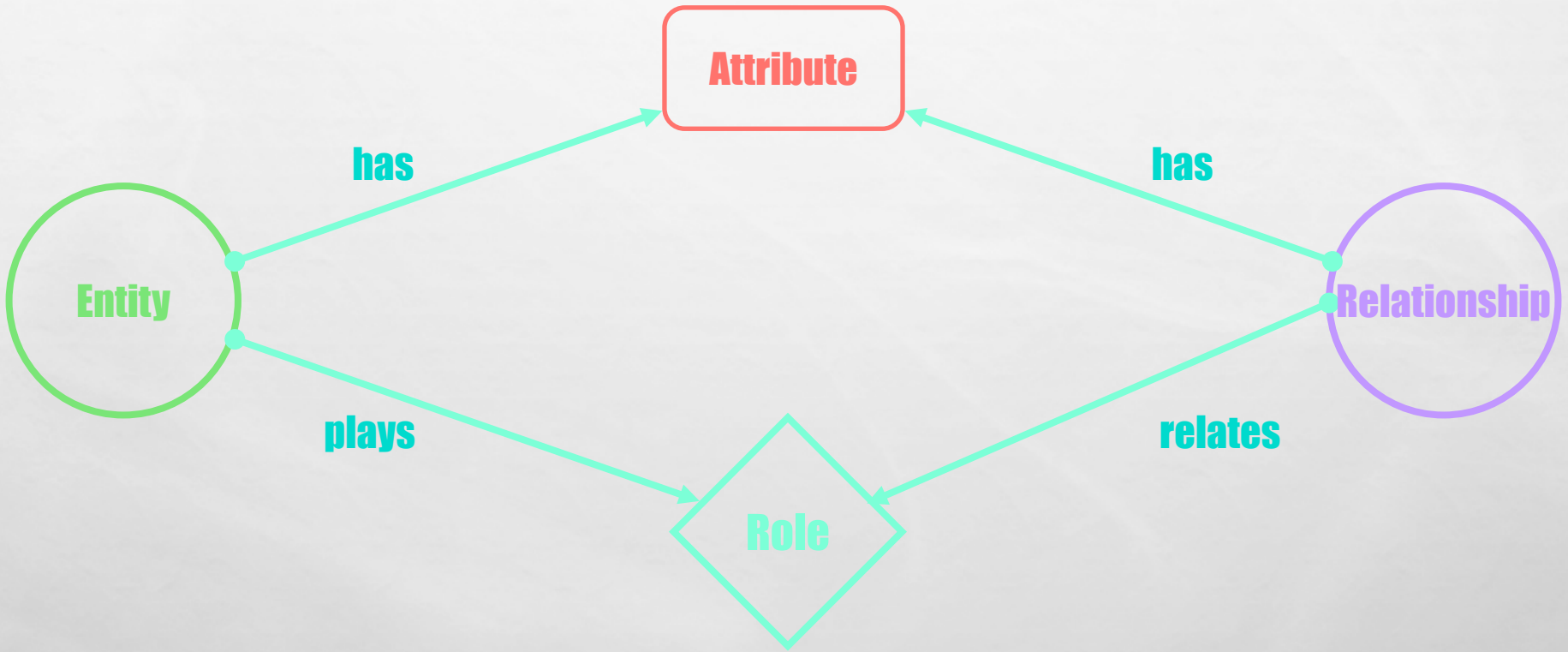
http://oc.acm.org/docs/7-12-2017%20-%20Sameer%20Singh%20-%20Explaining%20Black-Box%20ML%20Predictions.pdf

GRAPH DATABASES

GRAKN

Grakn is a distributed hyper-relational database for knowledge-oriented systems, i.e. a distributed knowledge base.

GRAKN KNOWLEDGE MODEL



GRAKN KNOWLEDGE MODEL

socrates, plato

teaches

e.g. teaches(socrates, plato)

mortal

e.g. mortal(socrates)

teacher, student

e.g. teaches(teacher: socrates, student: plato)

RULES IN GRAKN

HUMAN(X) -> MORTAL(X)

```
when {
        $x isa human
}, then {
        $x has mortal “true”
}
```

human *sub* mortal

BUILDING THE ML MODEL

...BACK TO MACHINE LEARNING

FEATURE ENGINEERING

- "GOOD" VS "BAD" COST MATRIX TAG
- BINARY DUMMY VARIABLES
- MODEL SELECTION AND OPTIMIZATION

PYTHON NOTEBOOK

localhost

GRAKN.AI - The Database for AI | GAdenis/SampleTeachCode.py at m... | Options | Home | KennyMLexploration | Seattle - Proto.io

jupyter MLexploration Last Checkpoint: Yesterday at 10:53 AM (autosaved) Logout

File Edit View Insert Cell Kernel Help Trusted Python 2

Code

In [41]:
```
ccMLmodel = BernoulliNB()

ccMLmodel.fit(train_data.values,train_labels.values.ravel())
predictions = ccMLmodel.predict(test_data.values)

print ccMLmodel
print "SKLearn calc accuracy:\t\t\t%.2f%%" % float(100*sklearn.metrics.accuracy_score(predictions,test_labels))
print "SKLearn calc accuracy (true Negative):\t%.2f%%" % float(100*sklearn.metrics.accuracy_score(predictions[test_labe
print "SKLearn calc accuracy (true Positive):\t%.2f%%" % float(100*sklearn.metrics.accuracy_score(predictions[test_labe
print "SKLearn calc f1 score:\t%.2f" % sklearn.metrics.f1_score(predictions,test_labels)
```

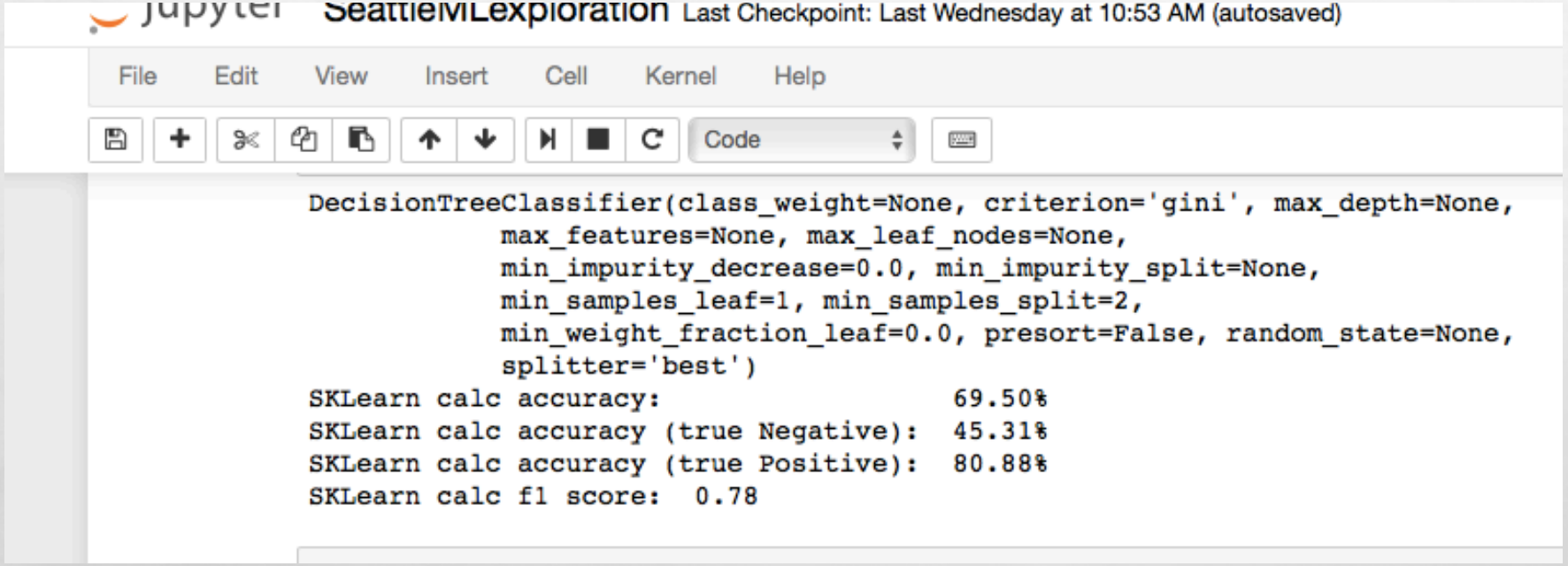
```
BernoulliNB(alpha=1.0, binarize=0.0, class_prior=None, fit_prior=True)
SKLearn calc accuracy:                  75.50%
SKLearn calc accuracy (true Negative):  54.69%
SKLearn calc accuracy (true Positive):  85.29%
SKLearn calc f1 score:  0.83
```

In [42]:
```
ccMLmodel = GaussianNB()

ccMLmodel.fit(train_data.values,train_labels.values.ravel())
predictions = ccMLmodel.predict(test_data.values)


print ccMLmodel
print "SKLearn calc accuracy:\t\t\t%.2f%%" % float(100*sklearn.metrics.accuracy_score(predictions,test_labels))
print "SKLearn calc accuracy (true Negative):\t%.2f%%" % float(100*sklearn.metrics.accuracy_score(predictions[test_labe
print "SKLearn calc accuracy (true Positive):\t%.2f%%" % float(100*sklearn.metrics.accuracy_score(predictions[test_labe
print "SKLearn calc f1 score:\t%.2f" % sklearn.metrics.f1_score(predictions,test_labels)
```

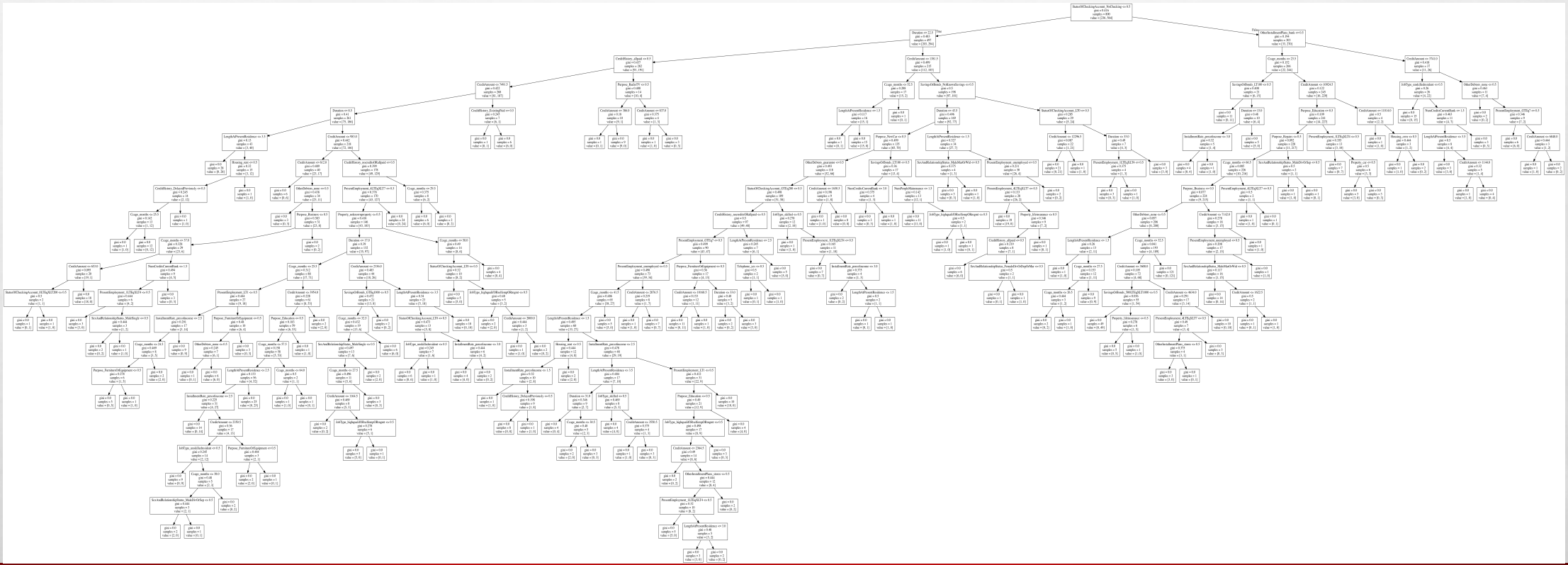
```
GaussianNB(priors=None)
SKLearn calc accuracy:                  70.50%
SKLearn calc accuracy (true Negative):  70.31%
SKLearn calc accuracy (true Positive):  70.59%
SKLearn calc f1 score:  0.76
```

DECISION & TREE INFORMATION GAIN

DECISION TREE CLASSIFIER



THE TREE



AWS ML

S3 Management Console | Amazon Machine Learning | eml-prod-emr.s3.amazonaws.com

https://console.aws.amazon.com/machinelearning/home?region=us-east-1#/predictor-insight/ml-r06DU3hE57S?tabId=realtimePre

Search

ML model report

- Summary
- Settings
- Monitoring

Tools

- Try real-time predictions

Evaluations

- Evaluation: ML mod...

Try real-time predictions

You submitted 20 out of 20 data values for this prediction.

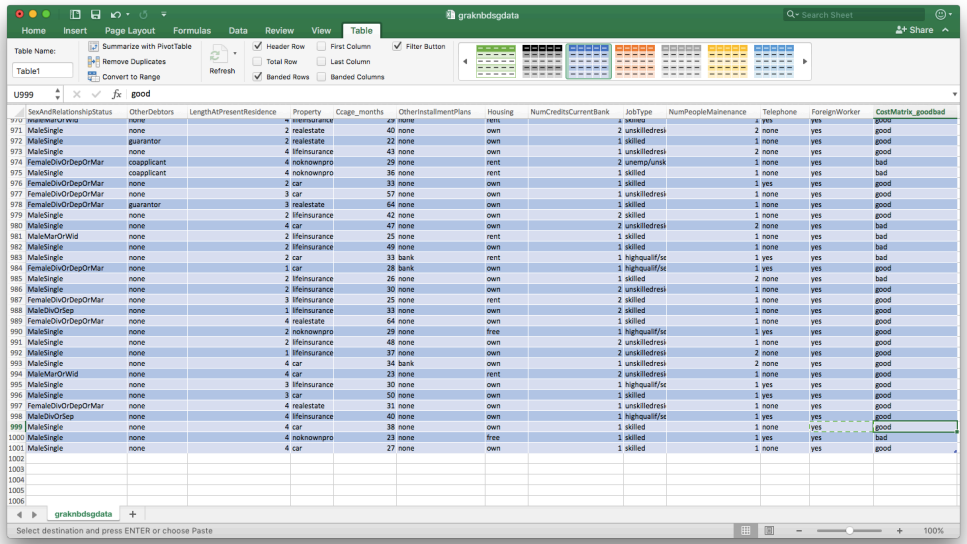
Try generating real-time predictions for free using the web browser on this page. To request a real-time prediction, complete the following form or provide a single data record in CSV format. To provide a data record, choose the **Paste a record** button. **Paste a record**

Attribute name

Items per page: 10 « ‹ 21 - 21 of 21 › »

	Name	Type	Value
21	CostMatrix_goodbad	Categorical	Target

« ‹ 21 - 21 of 21 › »



Clear data | Create prediction

Prediction results

Target name	CostMatrix_goodbad
ML model type	CATEGORICAL
Predicted class	good


```
{
  "Prediction": {
    "details": {
      "Algorithm": "SGD",
      "PredictiveModelType": "MULTICLASS"
    },
    "predictedLabel": "good",
    "predictedScores": {
      "bad": 0.05008869990706444,
      "good": 0.9499112963676453
    }
  }
}
```


Next steps

BUILDING THE GRAPH DB

HOW WOULD A GRAPH SOLUTION COMPARE?

CREDIT CLASSIFICATION

- Statlog (German Credit Data) Data Set
- Target attribute: good vs bad customer
- 20 attributes
  - Foreign worker? Telephone registered? Rented housing? Purpose? Credit amount?

CREDIT CLASSIFICATION: LEARNING

- AMIE+ Rule Learning algorithm
- Output: Horn clauses. They can be used directly in Grakn
  - condition_1, condition_2, ..., condition_n => (IMPLY) class

CREDIT CLASSIFICATION: CURRENT WORK

- The experiment had data translated into AMIE triples (denormalised)
- Extend the algorithm so it uses GRAQL queries!
- Extend the body conditions to include transitive relationships

RESULTS

MACHINE LEARNING RESULTS

- ~80% SCORE OVERALL ACCURACY IN PREDICTIONS
- SIMILAR PERFORMANCE ON AWS ML AND NAÏVE BAYES

GNB

```
print "SKLearn calc accuracy (true Negative):\t%.2f%%" % float(
print "SKLearn calc accuracy (true Positive):\t%.2f%%" % float(
print "SKLearn calc f1 score:\t%.2f" % sklearn.metrics.f1_score(
```

```
GaussianNB(priors=None)
SKLearn calc accuracy:                    70.50%
SKLearn calc accuracy (true Negative):    70.31%
SKLearn calc accuracy (true Positive):    70.59%
SKLearn calc f1 score:  0.76
```

BNB

```
print "SKLearn calc accuracy:\t\t\t%.2f%%" % float(100*sklearn.metrics.accuracy_score(predic
print "SKLearn calc accuracy (true Negative):\t%.2f%%" % float(100*sklearn.metrics.accuracy_
print "SKLearn calc accuracy (true Positive):\t%.2f%%" % float(100*sklearn.metrics.accuracy_
print "SKLearn calc f1 score:\t%.2f" % sklearn.metrics.f1_score(predictions,test_labels)
```

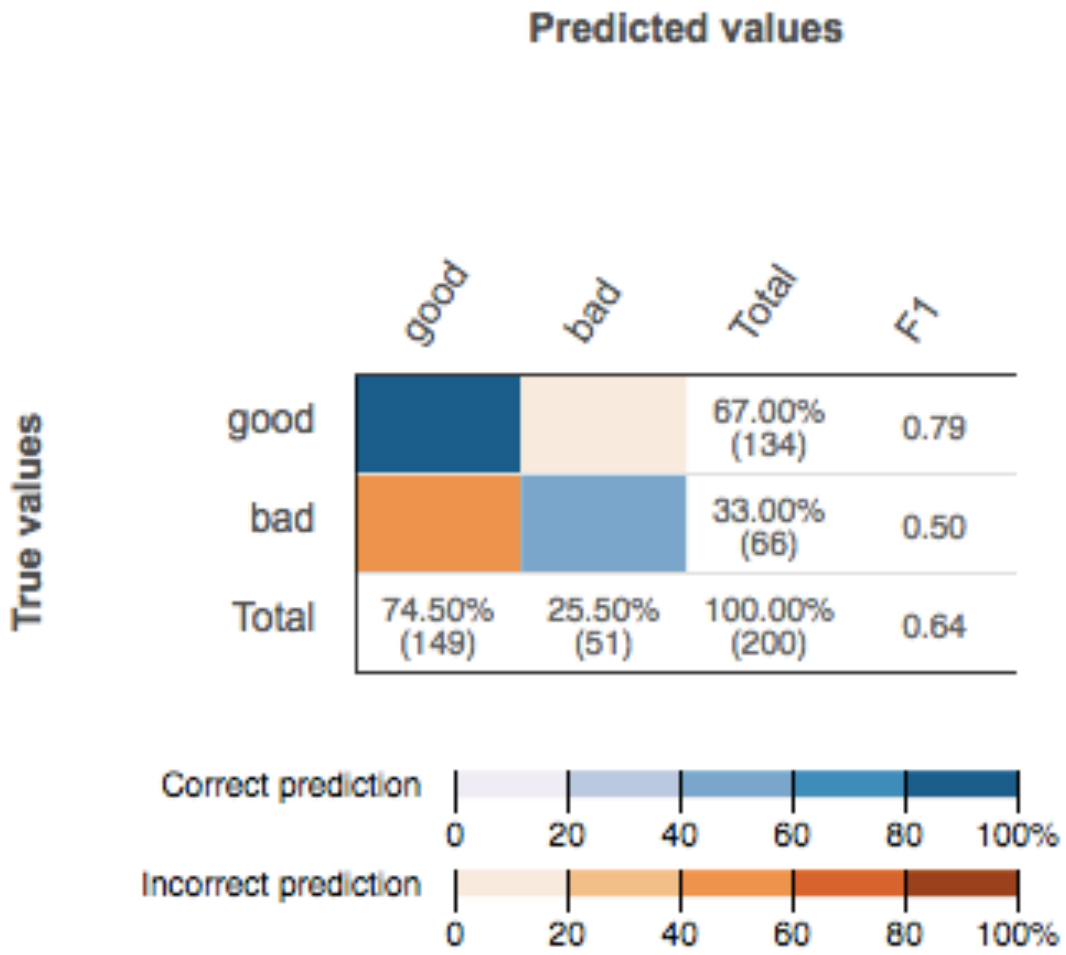
```
BernoulliNB(alpha=1.0, binarize=0.0, class_prior=None, fit_prior=True)
SKLearn calc accuracy:                    75.50%
SKLearn calc accuracy (true Negative):    54.69%
SKLearn calc accuracy (true Positive):    85.29%
SKLearn calc f1 score:    0.83
```

AWS ML PERFORMANCE

```
17/10/19 21:44:34 INFO:
17/10/19 21:44:34 INFO:
17/10/19 21:44:34 INFO: finished-training: total-passes=10 valid-records=7000 invalid-records=0
17/10/19 21:44:34 INFO: best learner:
17/10/19 21:44:34 INFO: learner-id=3535 model-configuration:       learning-rate=1.0
17/10/19 21:44:34 INFO: learner-id=3535 model-performance:         accuracy=0.8001 recall=0.7373 precision=0.7641 f1-score=0.7480
17/10/19 21:44:34 INFO: learner-id=3535 model-convergence:         negative-log-likelihood=4.669387e-01 (delta=1.000000e+00) is-converged=no
17/10/19 21:44:34 INFO: learner-id=3535 active-features:           updates=0000007000 min=0000021 max=0000021 mean=0000021 total-sum=0000147000
17/10/19 21:44:34 INFO: learner-id=3535 active-features-quantiles: quantile-10=0000021 quantile-50=0000021 quantile-90=0000022
17/10/19 21:44:34 INFO: learner-id=3535 model-status:              model-size=86112 (0.08 MB) #params=598 #pruning-calls=0000000000
17/10/19 21:44:34 INFO:
17/10/19 21:44:34 INFO:
17/10/19 21:44:34 INFO: final consolidation:
17/10/19 21:44:34 INFO: learner-id=3535 model-configuration:       learning-rate=1.0
17/10/19 21:44:34 INFO: learner-id=3535 model-convergence:         negative-log-likelihood=4.669387e-01 (delta=1.000000e+00) is-converged=no
17/10/19 21:44:34 INFO: learner-id=3535 active-features:           updates=0000007000 min=0000021 max=0000021 mean=0000021 total-sum=0000147000
17/10/19 21:44:34 INFO: learner-id=3535 active-features-quantiles: quantile-10=0000021 quantile-50=0000021 quantile-90=0000022
17/10/19 21:44:34 INFO: learner-id=3535 model-status:              model-size=86112 (0.08 MB) #params=598 #pruning-calls=0000000001
17/10/19 21:44:34 INFO:
17/10/19 21:44:34 INFO:
17/10/19 21:44:34 INFO: saved final learner:
17/10/19 21:44:34 INFO: learner-id=3535 model-configuration:       learning-rate=1.0
17/10/19 21:44:34 INFO: learner-id=3535 model-convergence:         negative-log-likelihood=4.669387e-01 (delta=1.000000e+00) is-converged=no
17/10/19 21:44:34 INFO: learner-id=3535 active-features:           updates=0000007000 min=0000021 max=0000021 mean=0000021 total-sum=0000147000
17/10/19 21:44:34 INFO: learner-id=3535 active-features-quantiles: quantile-10=0000021 quantile-50=0000021 quantile-90=0000022
17/10/19 21:44:34 INFO: learner-id=3535 model-status:              model-size=86112 (0.08 MB) #params=598 #pruning-calls=0000000001
17/10/19 21:44:34 INFO:
17/10/19 21:44:34 INFO: initial-training: finished
17/10/19 21:44:35 INFO: Foorprint size is 209675
17/10/19 21:44:35 INFO: Training complete.


------=_Part_326_758510164.1508449490015--
```

accu Highlight All Match Case Whole Words 81 of 81 matches Reached top of page, continued from bottom

80%


Predicted values
good
bad
Total
F1
True values
good
bad
Total
67.00%
(134)
0.79
33.00%
(66)
0.50
74.50%
(149)
25.50%
(51)
100.00%
(200)
0.64
Correct prediction
0
20
40
60
80
100%
Incorrect prediction
0
20
40
60
80
100%

GRAPH DB RESULTS

- $X HAS OTHER_PAYMENT_PLANS "NONE";
  $X HAS PURPOSE "RADIO/TV";
  $Y HAS VALUE "GOOD FLAG 1";
  0.799
- $X HAS CREDIT_HISTORY '"EXISTING PAID"';
  $W HAS IS_FOREIGN_WORKER "YES";
  ($X, $W) ISA HOLDS;
  $Y HAS VALUE "GOOD FLAG 2";
  0.672

Graph Results

- $x has other_payment_plans "none";
  $x has purpose "radio/tv";
  $y has value "good flag 1";
  0.799
- $x has credit_history '"existing paid"';
  $w has is_foreign_worker "yes";
  ($x, $w) isa holds;
  $y has value "good flag 2";
  0.672

RESULTS COMPARED

- ~75% TO 85% SCORE OVERALL ACCURACY IN PREDICTIONS
- ML SLIGHTLY BETTER PERFORMANCE, WITH ALL CASES PRESENT IN DATA
- GRAPH CAN MINE RULES, BUT EXPONENTIALLY INCREASING COMPLEXITY WITH NUMBER OF FEATURES

GRAPH DB CONCLUSIONS

- LEARNING FROM STRUCTURED DATA IS HARD! POTENTIALLY, ANYTHING IS RELATED TO ANYTHING
- WE NEED TO PICK A REASONABLE TRADEOFF BETWEEN COMPLEXITY OF THE LEARNING AND EXPRESSIVENESS OF THE LANGUAGE
- TRANSPARENT MODELS ARE HARDER TO TRAIN
- MIXED LEARNING TECHNIQUES ARE REALLY PROMISING
- THINK ABOUT THE LIFECYCLE OF THE MODEL

WHAT NOW?

WHAT DOES THE FUTURE HOLD?

INTEGRATING MACHINE LEARNING WITH GRAPH DATABASES

ONGOING COLLABORATION

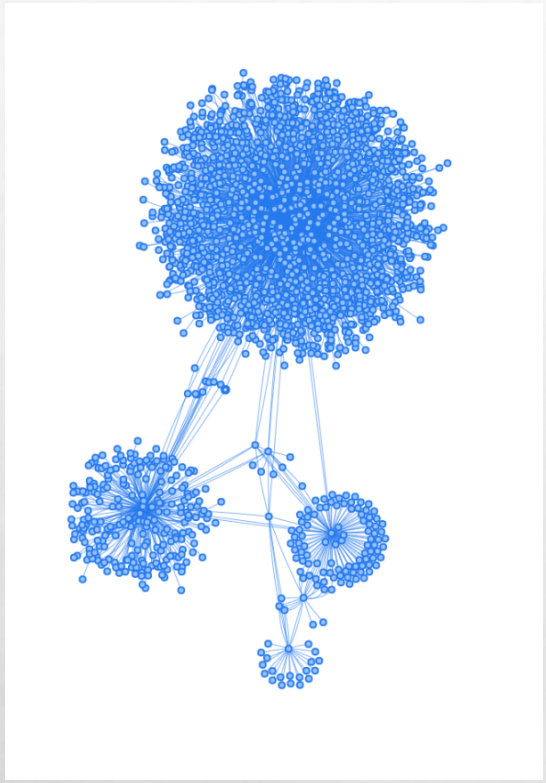
HTTP://WWW.BDS.GROUP

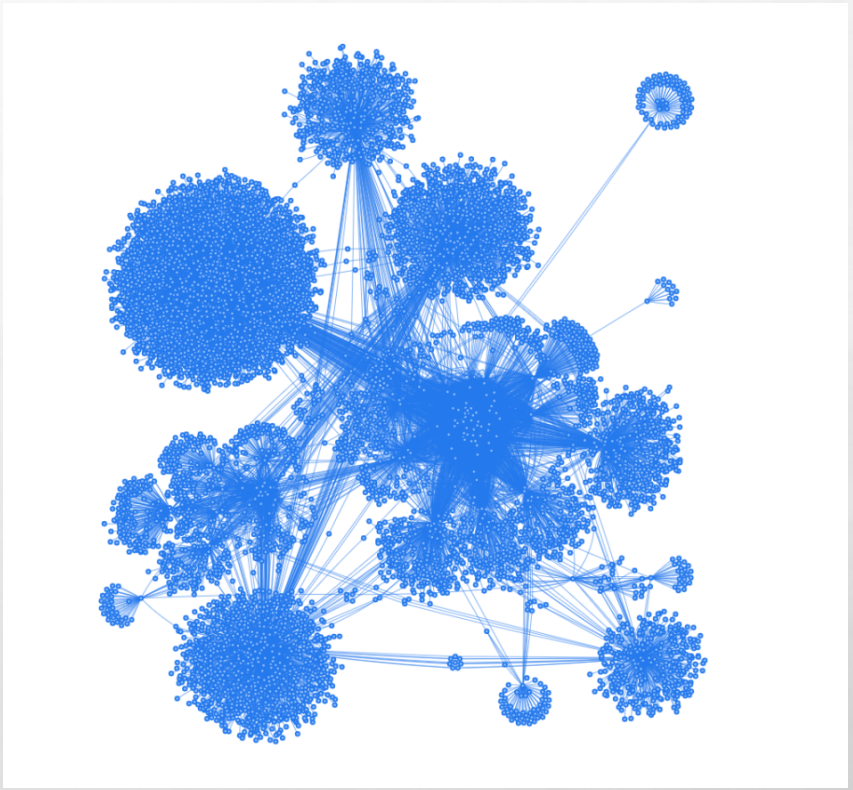
HTTP://WWW.GRAKN.AI

Applications and Analysis

- Characterizing and Representing Topologies
- Predicting Missing Edges
- DeConvoluting Overlapping Nodes (e.g., key="John Smith")
- Finding a Suitable Cost Function/Measure for ML in Graphs
- Identifying Outliers
- Classifying Connection Strengths

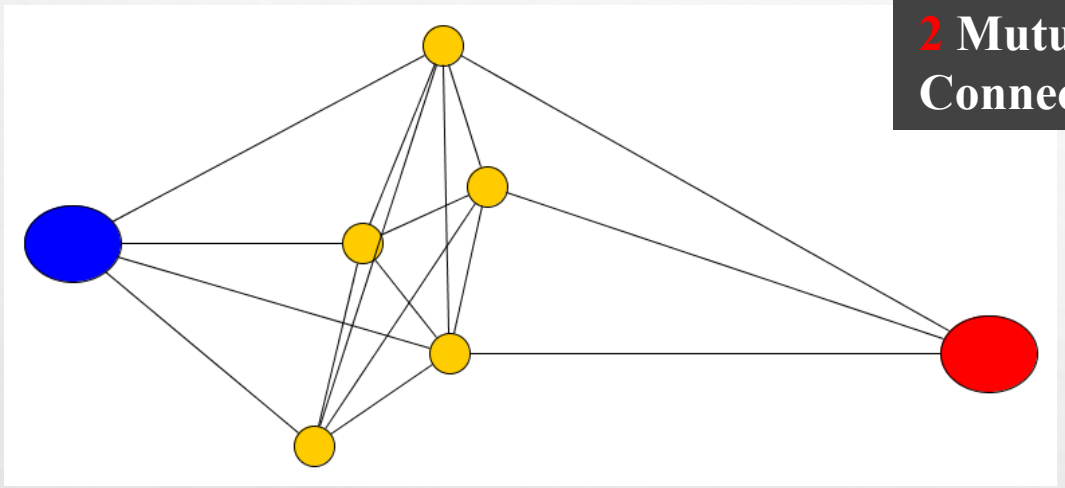
Characterizing Network Topologies



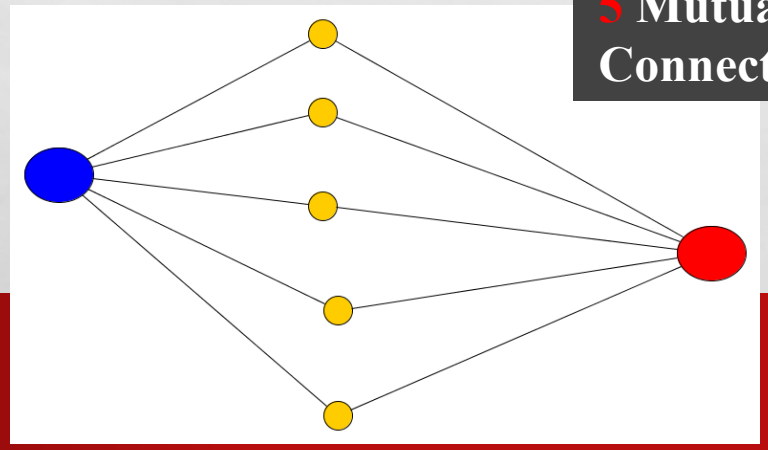


Is One of These an Outlier?

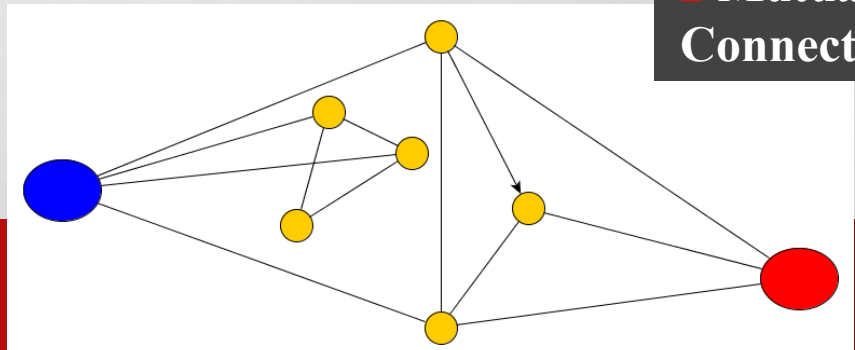
Which Ones Are Connected?



2 Mutual Connections



5 Mutual Connections



2 Mutual Connections

SEE SOME EXAMPLES AND PRODUCTS!

5:10 IN ROOM 407

THANK YOU

TECH BATTLE: GRAPHS VS MACHINE LEARNING

DENIS@BDS.GROUP

HAIKAL@GRAKN.AI

DENIS VRDOLJAK, BERKELEY DATA SCIENCE GROUP

HAIKAL PRIBADI, GRAKN AI