IR Experimentation

Jaime Arguello
INLS 509: Information Retrieval
jarguell@email.unc.edu

November 6, 2013

Outline

Test-collection evaluation (review)

Significance tests

Parameter Tuning

Cross-validation

Evaluation

- The main goal in experimental IR is to develop retrieval techniques that are better than the state of the art and to understand why they are better
- Basic question: Is system A better than system B?
- More often: Is system A with 'special sauce' better than system A without 'special sauce'?

Test Collection Evaluation

components

- Collection: a corpus of retrievable items or documents
- Topics: queries (input to system) and descriptions of what the hypothetical user is searching for
- Relevance judgements: a binary or graded indicator of relevance for each query-document pair
- Metrics: a measure of quality that operates on a ranking of known relevant and non-relevant documents

Test Collection Evaluation

queries

- Query 435: curbing population growth
- Description: What measures have been taken worldwide and what countries have been effective in curbing population growth? A relevant document must describe an actual case in which population measures have been taken and their results are known. Reduction measures must have been actively pursued. Passive events such as decease, which involuntarily reduce population, are not relevant.

(TREC 2005 HARD Track)

Test Collection Evaluation

metrics

- P@N
- R@N
- Average Precision (AP)
- Mean Reciprocal Rank (MRR)
- Normalized Discounted Cumulative Gain (NDCG)
- ....

Comparing Systems

P@10

Query	System A	System B
1	0.20	0.50
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
10	0.90	0.80
Average	0.41	0.48
	Difference	0.07

Significance Tests
motivation

- Why would it be risky to conclude that System B is better System A based on P@10?
- Put differently, what is it that we're trying to achieve?

Significance Tests
motivation

- In theory: the average performance of System B is greater than the average performance of System A for all possible queries!
- However, we don't have all queries. We have a sample (usually about 50).
- And, this sample may favor one system vs. the other!

Significance Tests
definition

- A significance test is a statistical tool that allows us to determine whether a difference in performance reflects a true pattern or just random chance

Significance Tests
ingredients

- Test statistic: a measure used to judge the two systems (e.g., the difference between their average P@10)
- Null hypothesis: no "true" difference between the two systems
- P-value: take the value of the observed test statistic and compute the probability of observing a value that large (or larger) under the null hypothesis

Significance Tests
ingredients

- If the p-value is large, we cannot reject the null hypothesis
- That is, we cannot claim that one system is better than the other

- If the p-value is small ($p<0.05$), we can reject the null hypothesis
- That is, the observed test-statistic is not due to random chance

Fisher's Randomization Test

procedure

- **Inputs:** counter = 0, N = 100,000
- Repeat N times:

  Step 1: for each query, flip a coin and if it lands 'heads', flip the result between System A and B

  Step 2: see whether the test statistic is equal to or greater than the one observed and, if so, increment counter

- **Output:** counter / N

Fisher's Randomization Test

Query	System A	System B
1	0.20	0.50
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
10	0.90	0.80
Average	0.41	0.48
	Difference	0.07

Fisher's Randomization Test

Query	System A	System B
1	**0.50**	**0.20**
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	**0.90**	**0.80**
7	0.30	0.10
8	0.10	0.20
9	**0.50**	**0.00**
10	0.90	0.80
Average	0.5	0.39
	Difference	-0.11

at least 0.07?

iteration = 1 counter = 0

Fisher's Randomization Test

Query	System A	System B
1	0.20	0.50
2	0.30	0.30
3	**0.10**	**0.10**
4	0.40	0.40
5	**1.00**	**1.00**
6	0.80	0.90
7	**0.10**	**0.30**
8	**0.20**	**0.10**
9	0.00	0.50
10	**0.08**	**0.90**
Average	0.318	0.5
	Difference	0.182

at least 0.07?

iteration = 2 counter = 1

Fisher's Randomization Test

Query	System A	System B
1	**0.50**	**0.20**
2	0.30	0.30
3	**0.10**	**0.10**
4	**0.40**	**0.40**
5	1.00	1.00
6	**0.90**	**0.80**
7	0.30	0.10
8	0.10	0.20
9	**0.50**	**0.00**
10	0.90	0.80
Average	0.5	0.39
	Difference	-0.11

at least 0.07?

iteration = 100,000 counter = 25,678

Fisher's Randomization Test
procedure

- **Inputs:** counter = 0, N = 100,000
- Repeat N times:

  Step 1: for each query, flip a coin and if it lands 'heads', flip the result between System A and B

  Step 2: see whether the test statistic is equal to or greater than the one observed and, if so, increment counter

- **Output:** counter / N = (25,678/100,00) = 0.25678

Fisher's Randomization Test
procedure

- Under the null hypothesis, the probability of observing a value of the test statistic of 0.07 or greater is about 0.26.
- Because $p > 0.05$, we cannot confidently say that the value of the test statistic is not due to random chance.
- A difference between the average P@10 values of 0.07 is not significant

Bootstrap Test
procedure

- **Inputs:** counter = 0, N = 100,000
- Repeat N times:

  Step 1: sample 10 queries (with replacement) from the set of 10 queries (called a subsample)

  Step 2: see whether the test statistic is equal to or greater than the one observed and, if so, increment counter

- **Output:** counter / N

Bootstrap Test

Query	System A	System B
1	0.20	0.50
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
10	0.90	0.80
Average	0.41	0.48
	Difference	0.07

Bootstrap Test

Query	System A	System B	**sample**
1	0.20	0.50	**0**
2	0.30	0.30	**1**
3	0.10	0.10	**2**
4	0.40	0.40	**2**
5	1.00	1.00	**0**
6	0.80	0.90	**1**
7	0.30	0.10	**1**
8	0.10	0.20	**1**
9	0.00	0.50	**2**
10	0.90	0.80	**0**

iteration = 1 counter = 0

Bootstrap Test

Query	System A	System B
2	0.30	0.30
3	0.10	0.10
3	0.10	0.10
4	0.40	0.40
4	0.40	0.40
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
9	0.00	0.50
Average	0.25	0.35
	Difference	0.1

iteration = 1 counter = 1

at least 0.07?

Bootstrap Test

Query	System A	System B	**sample**
1	0.20	0.50	**0**
2	0.30	0.30	**0**
3	0.10	0.10	**3**
4	0.40	0.40	**2**
5	1.00	1.00	**0**
6	0.80	0.90	**1**
7	0.30	0.10	**1**
8	0.10	0.20	**1**
9	0.00	0.50	**1**
10	0.90	0.80	**1**

iteration = 2 counter = 1

Bootstrap Test

Query	System A	System B
3	0.10	0.10
3	0.10	0.10
3	0.10	0.10
4	0.40	0.40
4	0.40	0.40
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
10	0.90	0.80
Average	0.32	0.36
	Difference	0.04

at least 0.07?

iteration = 2 counter = 1

Bootstrap Test

Query	System A	System B
1	0.20	0.50
1	0.20	0.50
4	0.40	0.40
4	0.40	0.40
4	0.40	0.40
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
8	0.10	0.20
10	0.90	0.80
Average	0.38	0.44
	Difference	0.06

at least 0.07?

iteration = 100,000 counter = 24,341

Bootstrap Test
procedure

- **Inputs:** counter = 0, N = 100,000
- Repeat N times:

  Step 1: sample 10 queries (with replacement) from the set of 10 queries (called a subsample)

  Step 2: see whether the test statistic is equal to or greater than the one observed and, if so, increment counter

- **Output:** counter / N = (24,341/100,00) = 0.24341

Significance Tests

summary

- Significance tests help us determine whether the outcome of an experiment signals a “true” trend
- The null hypothesis is that the observed outcome is due to random chance (sample bias, error, etc.)
- There are many types of tests
- Parametric tests: assume a particular distribution for the test statistic under the null hypothesis
- Non-parametric tests: make no assumptions about the test statistic distribution under the null hypothesis
- The randomization and bootstrap tests make no assumptions, are robust, and easy to understand

Comparing Systems

parameter tuning

Query	System A	System B
1	0.20	0.50
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
10	0.90	0.80
Average	0.41	0.48
	Difference	0.07

Parameter Tuning
motivation

- Search algorithms have lots of moving parts
- We can think of these parameters as “knobs” that need to be tweaked or tuned
- The goal is to set these parameter values such that we maximize performance
- We need to do this for both systems, not just the one we want to win!
- Can you think of some example parameters?

Parameter Tuning

- Query-likelihood model with linear interpolation

$$score(Q, D) = \prod_{q \in Q} (\lambda P(q|\theta_D) + (1 - \lambda) P(q|\theta_C))$$

- Parameter λ avoids zero probabilities when a document is missing a query-term
- How should we determine the value of λ ?

Parameter Tuning

- How should we determine the value of λ?
- Option -1: roll the dice, close your eyes, and hope for the best
- Option 0: take a conservative guess (e.g., $\lambda = 0.5$)?
- Option 1: try out a range of values (e.g., $\lambda = 0.0, 0.1, 0.2, \ldots, 1.0$) and set it to the value that maximizes performance based on a sensible metric?

Parameter Tuning

λ =	Average P@10 =
0.0	0.25
0.1	0.27
0.2	0.29
0.3	0.35
0.4	0.45
0.5	0.50
0.6	**0.55**
0.7	0.47
0.8	0.35
0.9	0.20
1.0	0.00

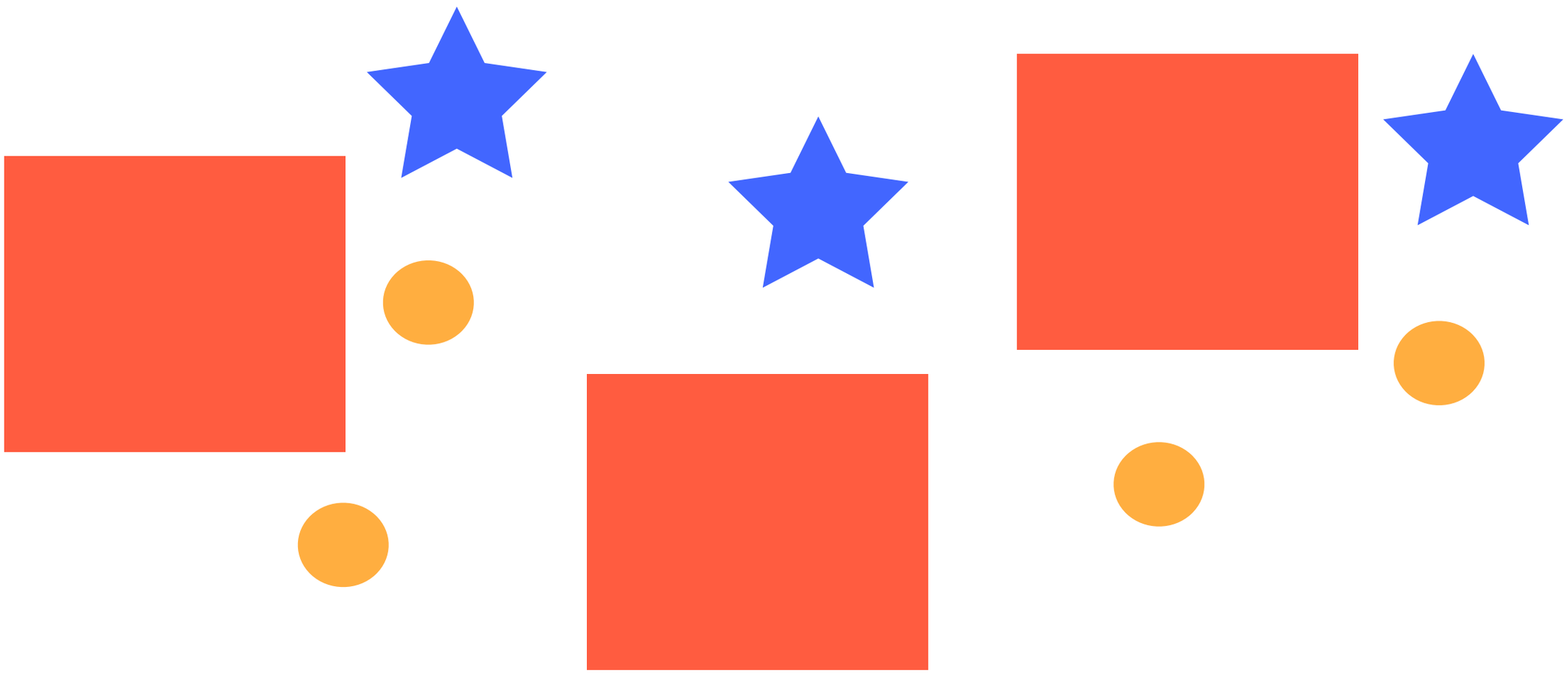
50 TEST QUERIES

How well will the QL model do after parameter tuning?

Parameter Tuning

toy example

- Objective: distinguish between stars, squares, and circles



- Parameters: the relative importance between (1) size, (2) color, and (3) number of sides

Parameter Tuning

λ =	Average P@10 =
0.0	0.25
0.1	0.27
0.2	0.29
0.3	0.35
0.4	0.45
0.5	0.50
0.6	**0.55**
0.7	0.47
0.8	0.35
0.9	0.20
1.0	0.00

50 TEST QUERIES

Why is this a bad idea?

Parameter Tuning

- The goal is to set parameter values such that we maximize performance
- What is the performance that we are really interested in?
- We care about previously unseen queries
- We care about generalization performance!
- Our sample of queries may contain regularities that are not meaningful
- We care about those regularities that are meaningful for the overall population!

Parameter Tuning

System A

Average
P@10 = 0.60

System A
+ ‘special sauce’

Average
P@10 = 0.55

Parameter Tuning

- Option 2:

  1. divide the set of 50 queries into two sets:

     - training set: a set of queries used to find the best parameter values (e.g., 40 queries)

     - test set: a held-out set used to evaluate model performance (e.g., 10 queries)

  2. train: find the parameter value that maximizes performance on the training set

  3. test: evaluate the model (with the best training-set parameter value) on the test set

Parameter Tuning

Parameter Tuning

- Split the data into two sets.
- Find the parameter value that maximize performance on the training set.
- Evaluate the system with that parameter value on the test set.

λ = 0.6

P@10 = 0.50

Parameter Tuning

- Split the data into two sets.
- Find the parameter value that maximize performance on the training set.
- Evaluate the system with that parameter value on the test set.

λ = 0.6

P@10 = 0.50

Advantages and Disadvantages?

Single Train/Test Split

- Advantage
  - the data used to find the optimal parameter value is not the same data used to test!
  - we are testing generalization performance.
- Disadvantage
  - we are putting all our eggs in one basket!
  - out of pure coincidence, the training set may have regularities that don't generalize to the test set

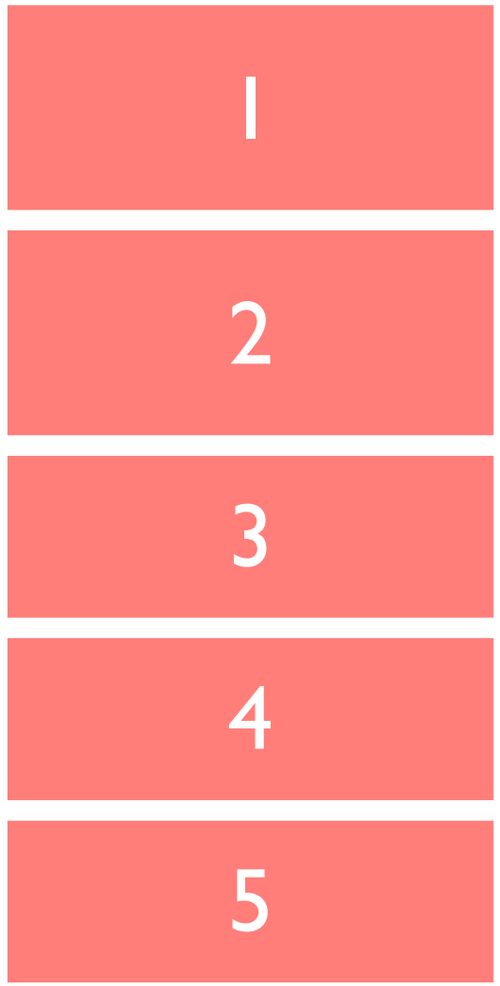
Parameter Tuning

- Option 3: cross-validation
  1. divide the set of 50 queries into N sets of 50/N queries
  2. use the union of N-1 sets to find the best parameter values
  3. measure performance (using the best parameters) on the held-out set
  4. do steps 2-3 N times
  5. average performance across the N held-out sets
- This is called N-fold cross-validation (usually, N=10)

Cross-Validation

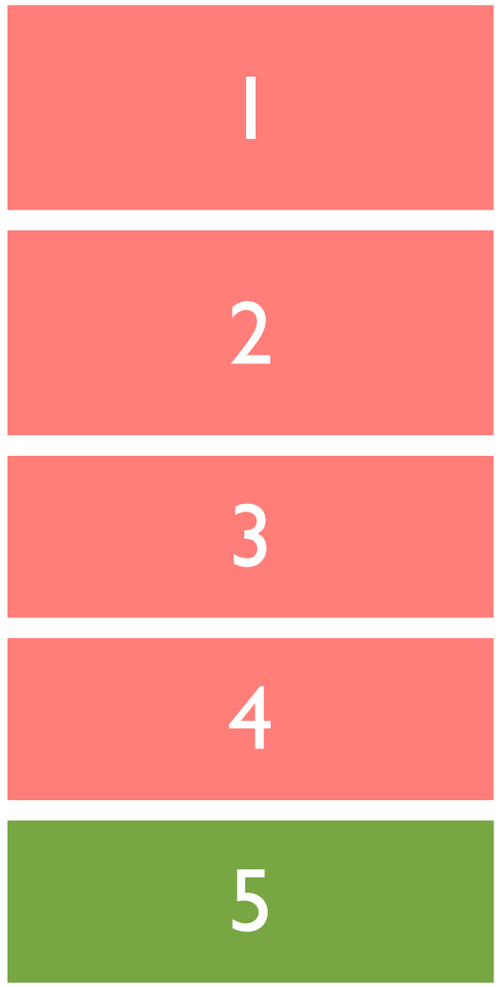
Cross-Validation

- Split the data into N = 5 folds of 10 queries each



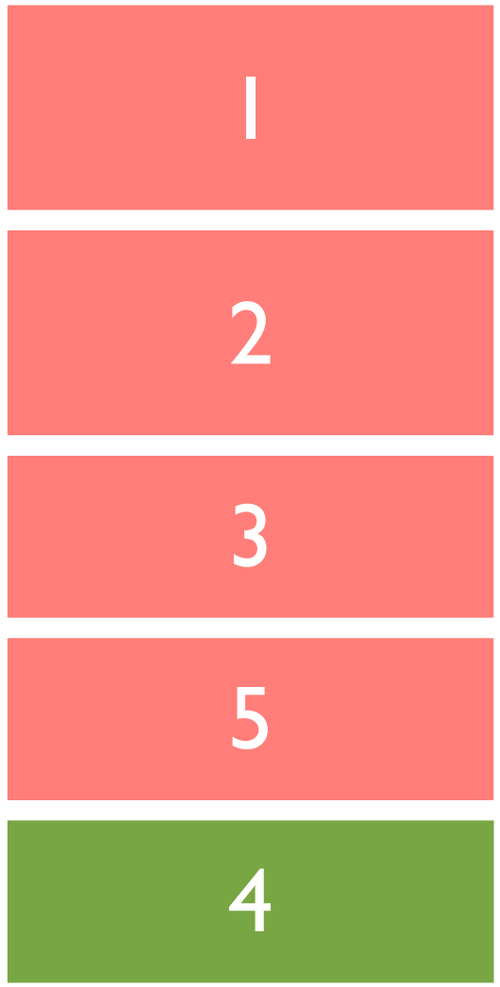
Cross-Validation

- For each fold, find the parameter value that maximizes performance on the union of N - 1 folds and test this parameter value on the held-out fold.



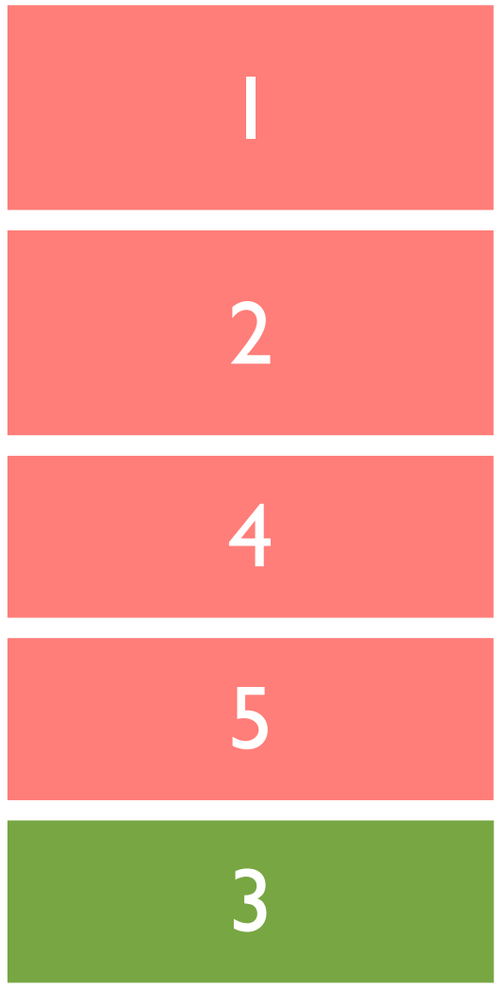
Cross-Validation

- For each fold, find the parameter value that maximizes performance on the union of N - 1 folds and test this parameter value on the held-out fold.



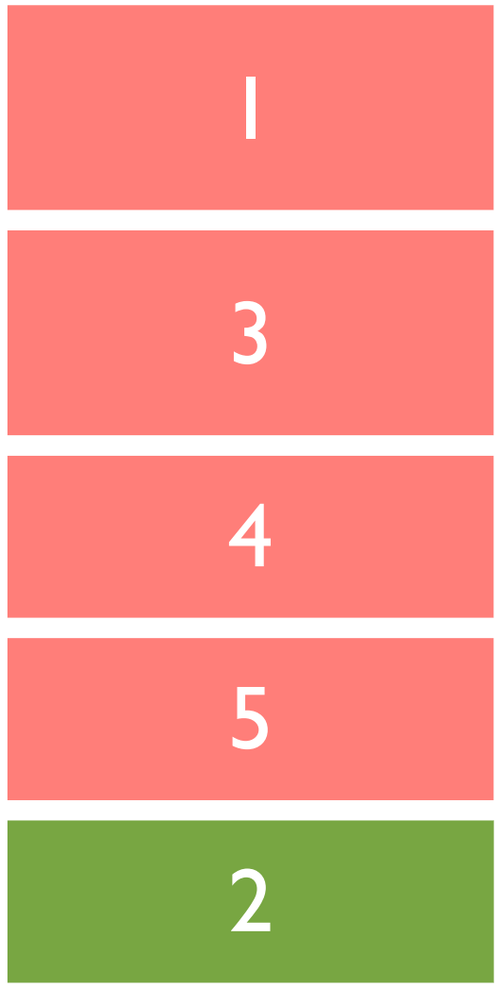
Cross-Validation

- For each fold, find the parameter value that maximizes performance on the union of N - 1 folds and test this parameter value on the held-out fold.



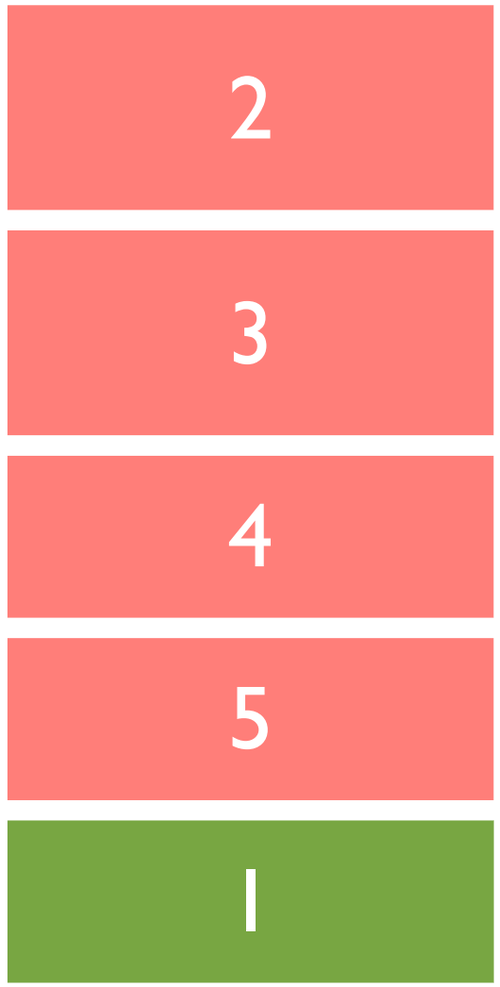
Cross-Validation

- For each fold, find the parameter value that maximizes performance on the union of N - 1 folds and test this parameter value on the held-out fold.



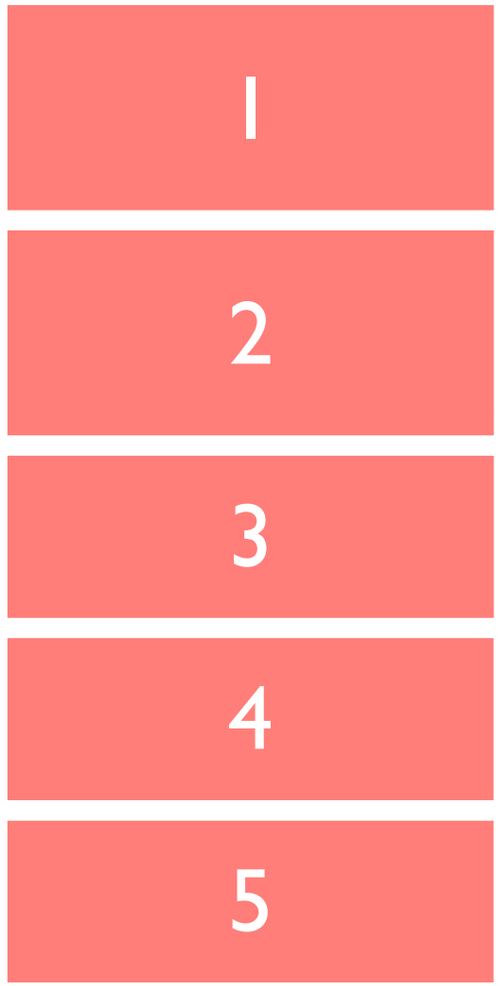
Cross-Validation

- For each fold, find the parameter value that maximizes performance on the union of N - 1 folds and test this parameter value on the held-out fold.



Cross-Validation

- Average the performance across held-out folds



P@10 = 0.80

P@10 = 0.50

P@10 = 0.70

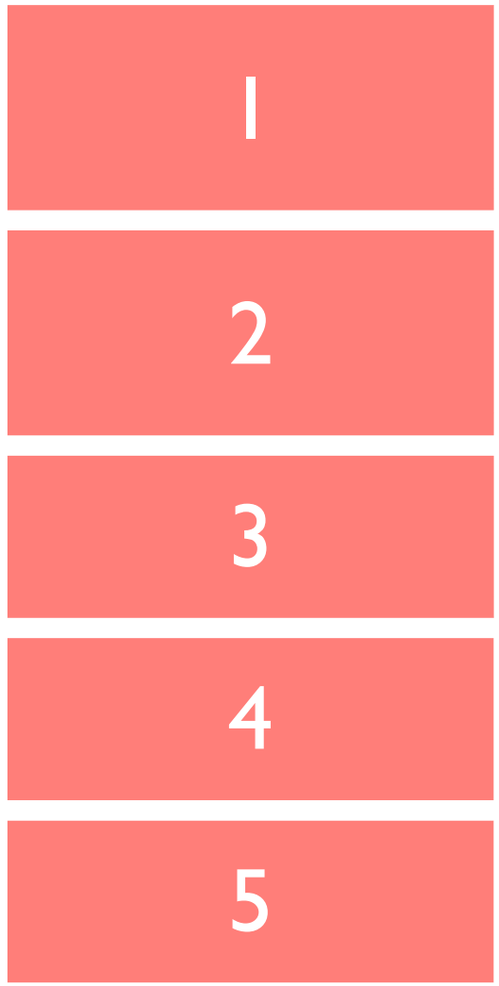
P@10 = 0.55

P@10 = 0.50

Average **P@10 = 0.61**

Cross-Validation

- Average the performance across held-out folds



P@10 = 0.80

P@10 = 0.50

P@10 = 0.70

P@10 = 0.55

P@10 = 0.50

Average **P@10 = 0.61**

Advantages and Disadvantages?

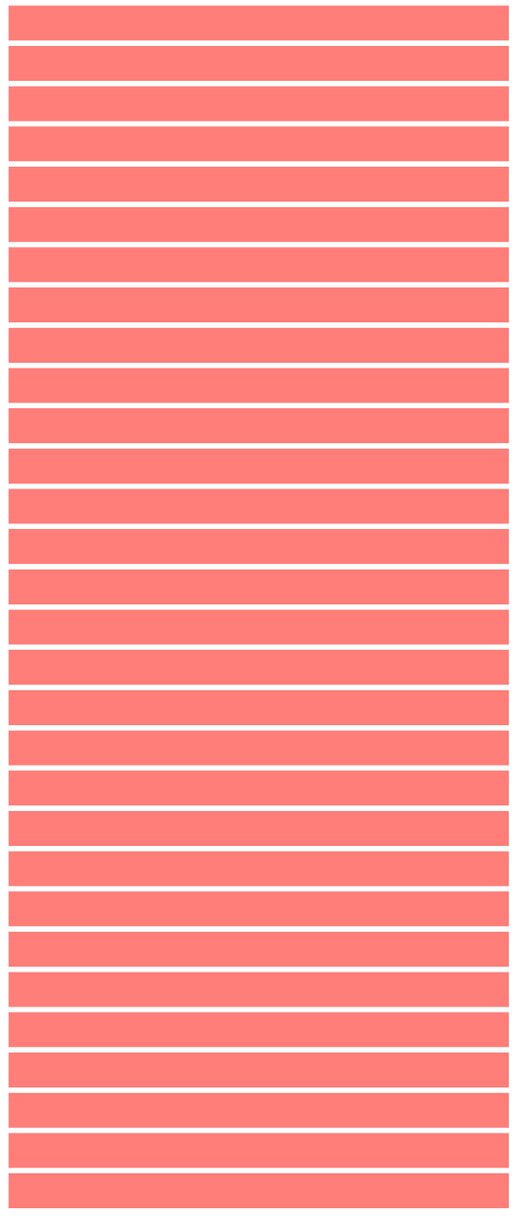
N-Fold Cross-Validation

- Advantage
  - multiple rounds of generalization performance.
- Disadvantage
  - ultimately, we'll tune parameters on the set of 50 queries and send our system into the world.
  - a model trained on 50 queries should perform better than one trained on 40.
  - thus, we may be underestimating the model's performance!

Leave-One-Out Cross-Validation

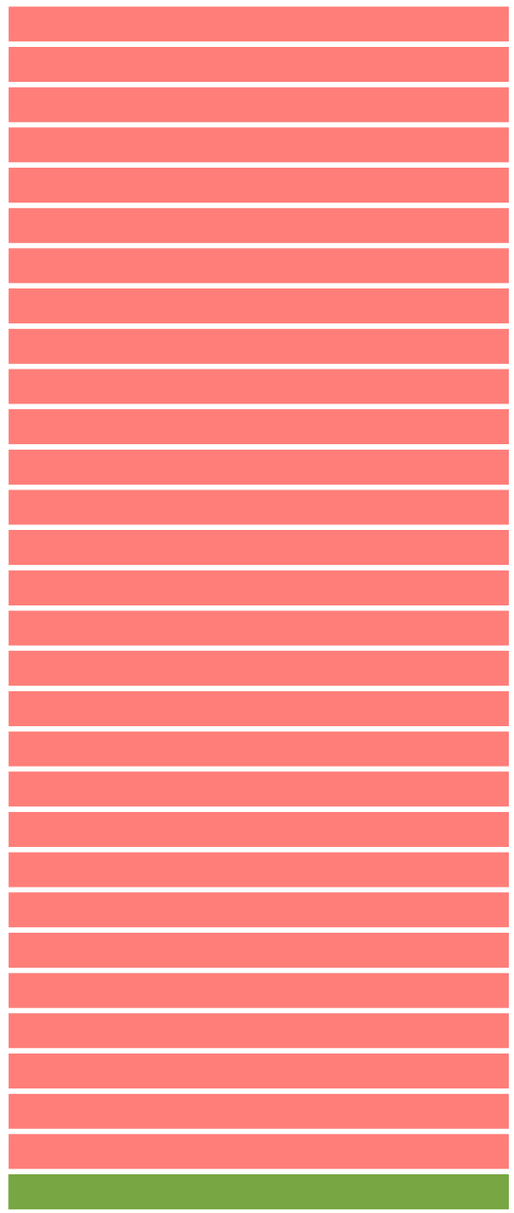
Leave-One-Out Cross-Validation

- Split the data into N = 50 folds of 1 queries each



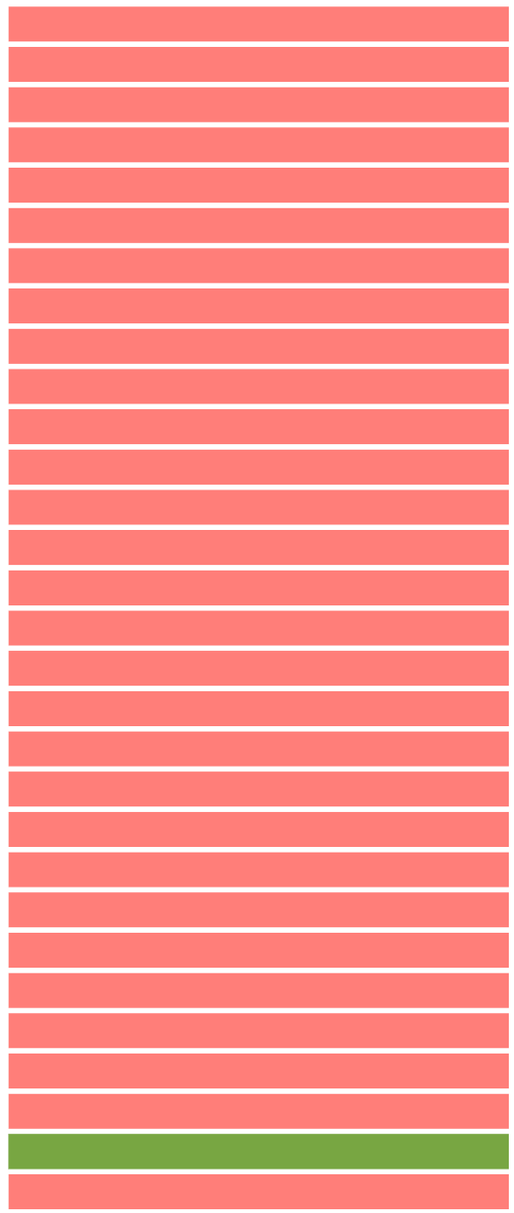
Leave-One-Out Cross-Validation

- For each query, find the parameter value that maximize performance on for the other queries and and test (using this parameter value) on the held-out query.



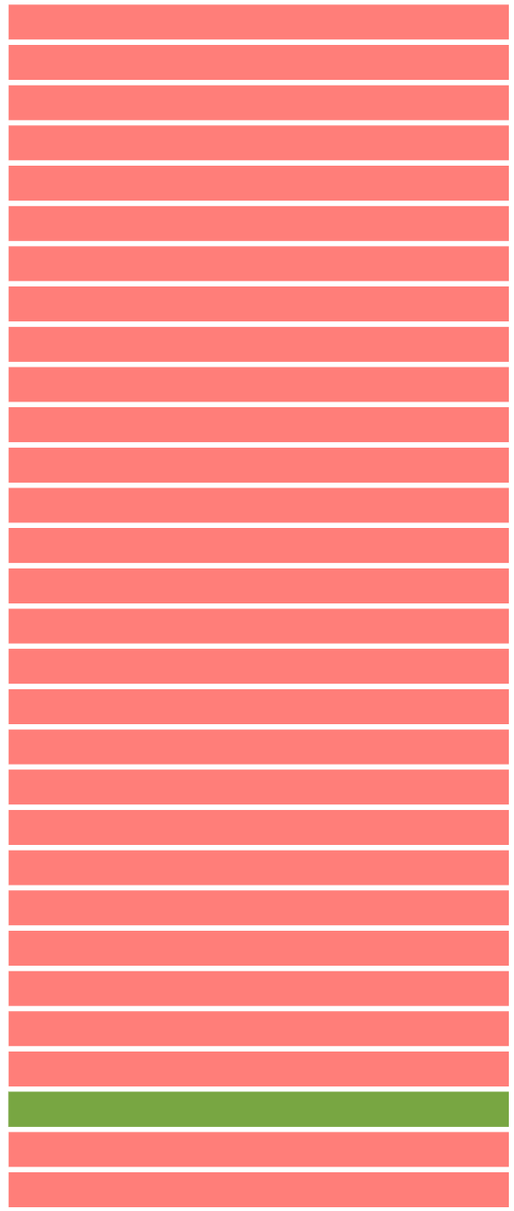
Leave-One-Out Cross-Validation

- For each query, find the parameter value that maximize performance on for the other queries and and test (using this parameter value) on the held-out query.



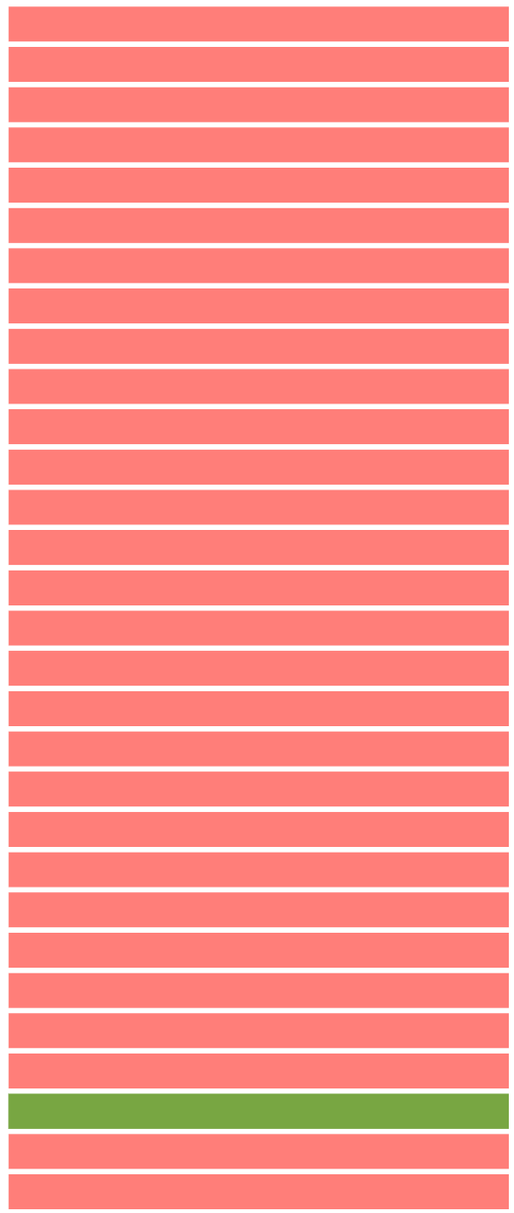
Leave-One-Out Cross-Validation

- For each query, find the parameter value that maximize performance on for the other queries and and test (using this parameter value) on the held-out query.
- And so on ...
- Finally, average the performance for each held-out query



Leave-One-Out Cross-Validation

- For each query, find the parameter value that maximize performance on for the other queries and and test (using this parameter value) on the held-out query.
- And so on ...
- Finally, average the performance for each held-out query



Advantages and Disadvantages?

Leave-One-Out Cross-Validation

- Advantages
  - multiple rounds of generalization performance.
  - each training fold is as similar as possible to the one we will ultimately use to tune parameters before sending the system out into the world.
- Disadvantage
  - our estimate of generalization performance may still be artificially high
  - why?

Leave-One-Out Cross-Validation

- Advantages
  - multiple rounds of generalization performance.
  - each training fold is as similar as possible to the one we will ultimately use to tune parameters before sending the system out into the world.
- Disadvantage
  - our estimate of generalization performance may still be artificially high
  - we are likely to try lots of different things and pick the one with the best "generalization" performance
  - still indirectly over-training to the dataset (sigh...)

Putting it all Together

- For each system, tune and test the necessary parameters using N-fold cross-validation
- Use the same folds for both systems
- Compare the difference in average performance across folds using a significance test

Fold	System A	System B
1	0.20	0.50
2	0.30	0.30
3	0.10	0.10
4	0.40	0.40
5	1.00	1.00
6	0.80	0.90
7	0.30	0.10
8	0.10	0.20
9	0.00	0.50
10	0.90	0.80
Average	0.41	0.48
	Difference	0.07

The Annoying Details
lots of experiments

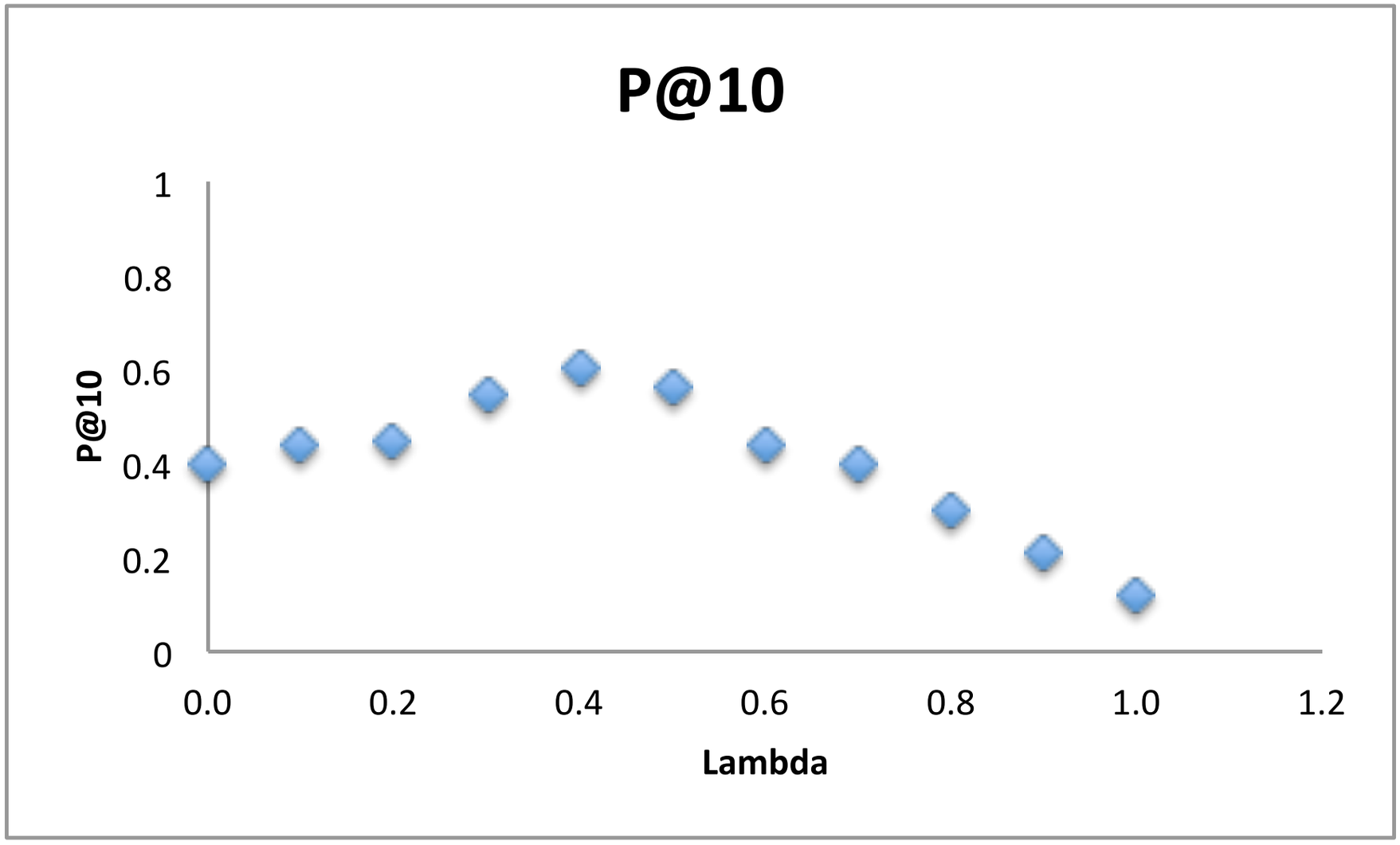
- A model with three parameters (each with a range between 0.0 and 1.0) has 10 x 10 x 10 = 1000 parameter combinations
- With 10-fold cross-validation, that's (10,010 x 2) = 20,020 batch evaluation cycles
- You can't do this on your iPad.

The Annoying Details
resisting temptation

- If your goal is to outperform some baseline system, it can be tempting to not tune the baseline.
- For example, if the optimal parameter value is always the same, you might need to increase the granularity of your parameter search.

The Annoying Details

resisting temptation



- Suppose that the optimal value of Lambda for different training folds is consistently 0.40.
- Are we done?

Summary

- Statistic significance tests can help us associate a confidence value that our observation (necessarily limited in scope) generalizes to space of all possible queries.
- Tune parameters. For the baseline system and for the experimental system.
- Resist temptation and be skeptic and thorough.
- Enjoy the process! Remember your junior-high scientific method lectures.