CSE 519: Data Science
Steven Skiena
Stony Brook University

Lecture 14: Validating Models

How Good is Your Model?

After you train a model, you need to evaluate it on your testing data.

What statistics are most meaningful for:

- Classification models (which produce labels)
- Regression models (which produce numerical value predictions)

Evaluating Classifiers

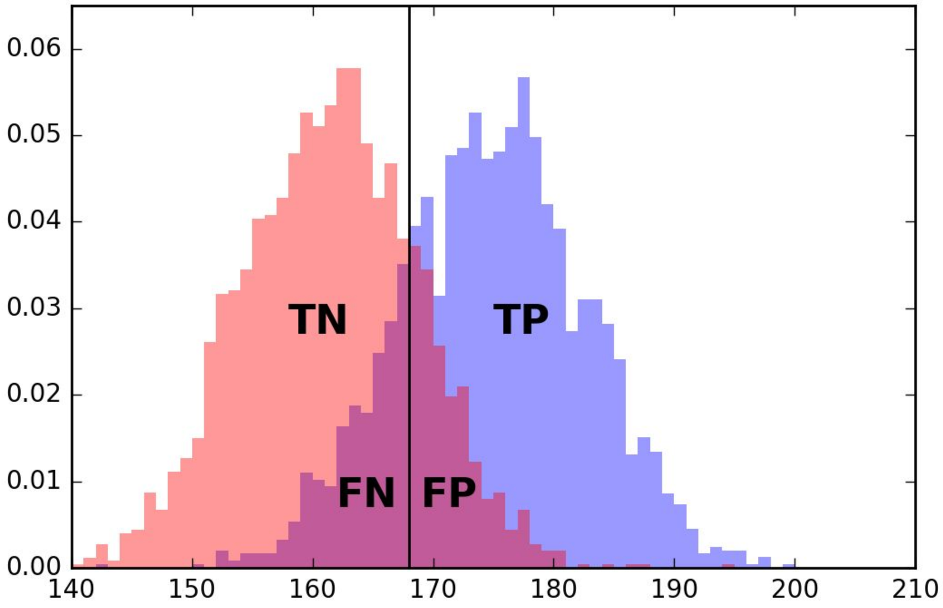
		Predicted Class	
		Yes	No
Actual Class	Yes	TP	FN
	No	FP	TN

There are four possible outcomes for a binary classifier:

- True positives (TP) where + is labeled +
- True negative (TN) where - is labeled -
- False positives (FP) where - is labeled +
- False negatives (FN) where + is labeled -

Threshold Classifiers

Identifying the best threshold requires deciding on an appropriate evaluation metric.



Accuracy

The accuracy is the ratio of correct predictions over total predictions:

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP}$$

The monkey would randomly guess positive with p=0.5, with accuracy 50%.

Picking the biggest class yields >=50%.

Precision

When |*P*|<<|*N*|, accuracy is a silly measure.

If only 5% of tests say cancer, are we happy with a 50% accurate monkey?

$$precision = \frac{TP}{(TP + FP)}$$

The monkey would achieve 5% precision, as would a sharp always saying cancer.

Recall

In the cancer case, we would tolerate some false positive (scares) to identify real cases:

$$recall = \frac{TP}{(TP + FN)}$$

Recall measures being right on positive instances.

Saying everyone has cancer gives perfect recall!

F-Score

To get a meaningful single score balancing precision and recall use F-score:

$$F = 2 \cdot \frac{\mathrm{precision} \cdot \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}$$

The harmonic mean is always less than/equal to the arithmetic mean, making it tough to get a high F-score.

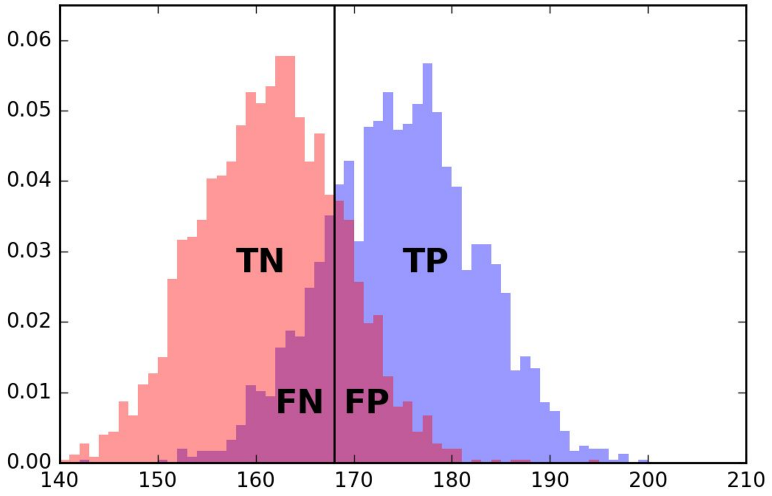
Take Away Lessons

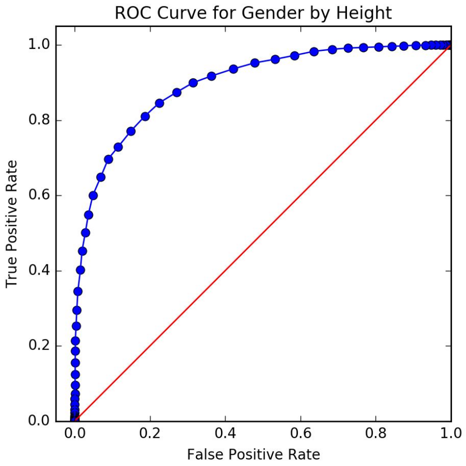
- Accuracy is misleading when the class sizes are substantially different.
- High precision is very hard to achieve in unbalanced class sizes.
- F-score does the best job of any single statistic, but all four work together to describe the performance of a classifier.

Receiver-Operator (ROC) Curves

Varying the threshold changes recall/precision.

Area under ROC is a measure of accuracy.





Evaluating Multiclass Systems

Classification gets harder with more classes.

The confusion matrix shows where the mistakes are being made: 5->3, 8->2

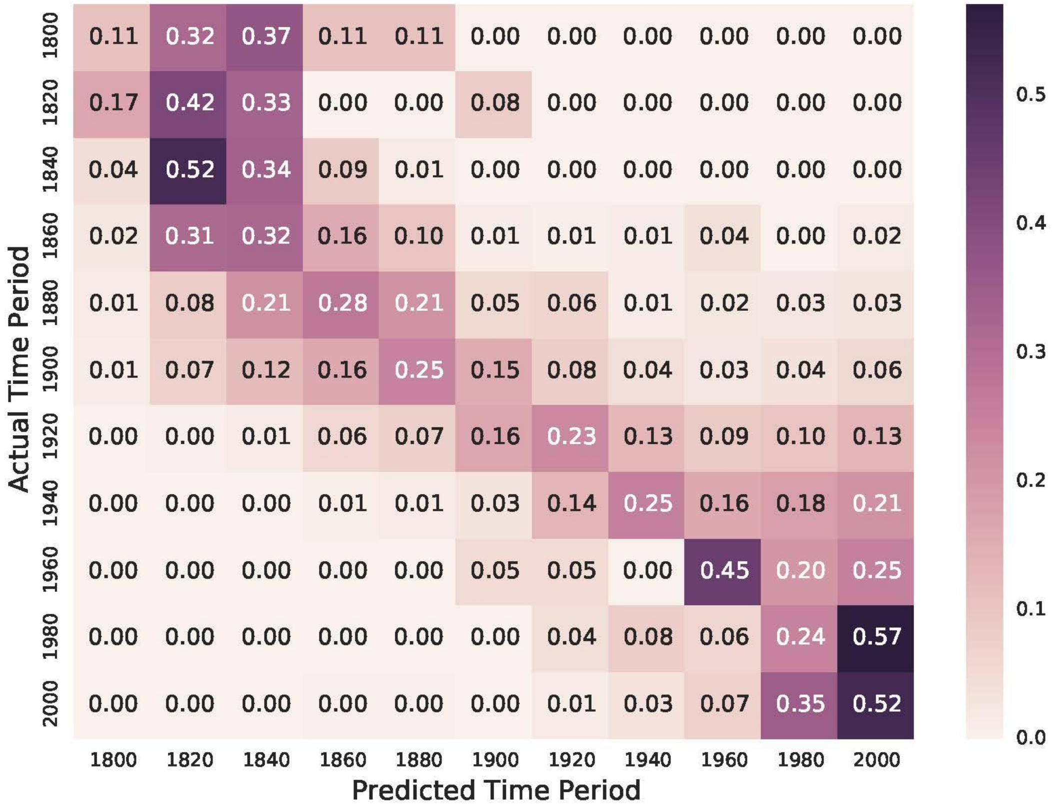
Digits	0	1	2	3	4	5	6	7	8	9
0	351	0	5	4	2	7	2	1	6	0
1	0	254	0	0	2	0	0	1	1	2
2	1	1	166	4	5	1	3	2	2	1
3	1	2	4	142	0	5	0	1	4	0
4	3	3	8	1	180	3	2	5	4	4
5	0	0	3	11	0	140	3	0	7	1
6	0	2	2	0	4	0	158	0	1	0
7	0	0	2	2	1	0	0	132	2	1
8	2	1	8	0	0	0	2	1	137	1
9	1	1	0	2	6	4	0	4	2	167

Figure 7.7: Confusion matrix for digits in a zip code OCR program.

Confusion Matrix: Dating Documents

What periods are most often confused with each other?

The main diagonal is not exactly where the heaviest weight always is.



Scoring Hard Problems Easier

Too low a classification rate is discouraging and often misleading with multiple classes.

The *top-k* success rate gives you credit if the right label would have been one of your first *k* guesses.

It is important to pick *k* so that real improvements can be recognized.

Summary Statistics: Numerical Error

For numerical values, error is a function of the delta between forecast *f* and observation *o*:

- Absolute error: *(f - o)*
- Relative error: *(f - o) / o* (typically better)

These can be aggregated over many tests:

- Mean or median squared error $\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y}_i - Y_i)^2.$
- Root mean squared error $\mathrm{RMSD}(\hat{\theta}) = \sqrt{\mathrm{MSE}(\hat{\theta})} = \sqrt{\mathrm{E}((\hat{\theta} - \theta)^2)}.$

Evaluation Data

The best way to assess models involve out-of-sample predictions, results on data you never saw (or even better did not exist) when you built the model.

Partitioning the input into training (60%), testing (20%) and evaluation (20%) data works only if you never open evaluation data until the end.

Sins in Evaluation

Formal evaluation metrics reduce models to a few summary statistics.

But many problems can be hidden by statistics:

- Did I mix training and evaluation data?
- Do I have bugs in my implementation?

Revealing such errors requires understanding the types of errors your model makes.

Building an Evaluation Environment

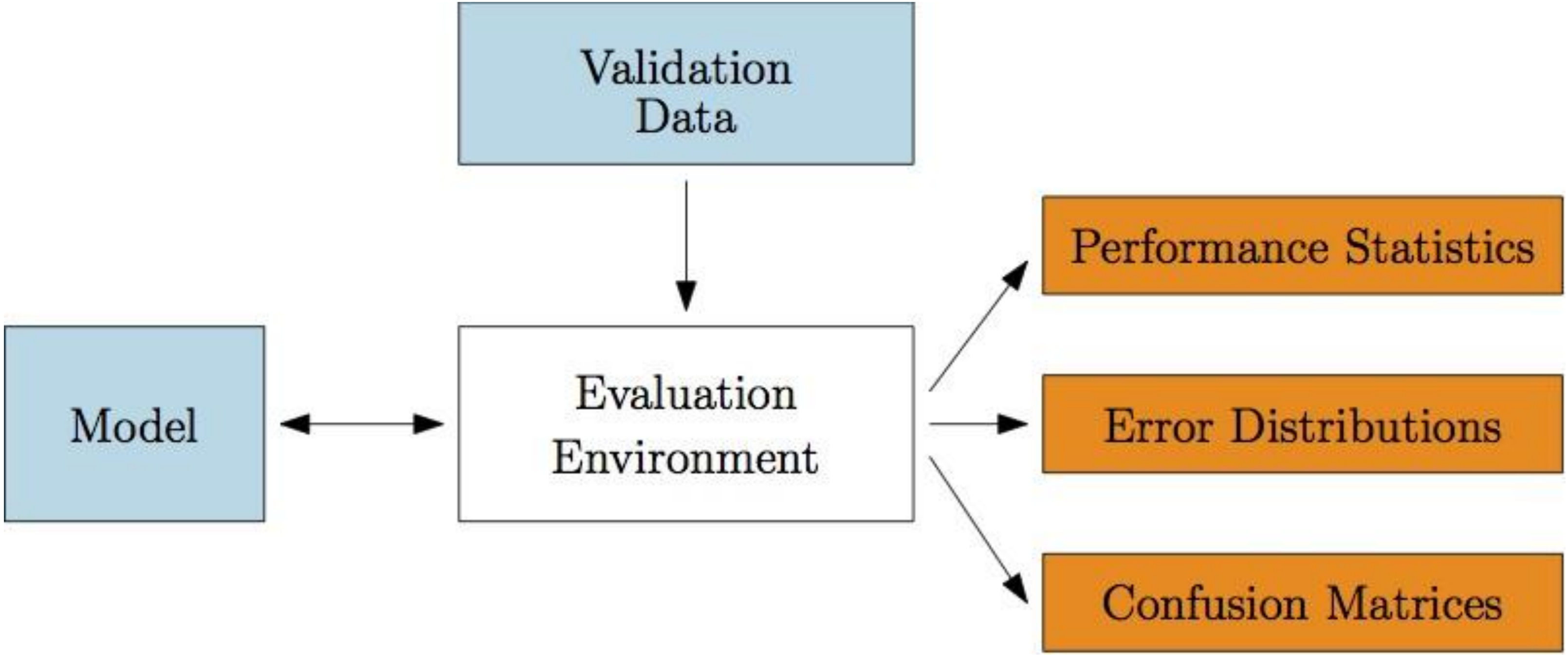
You need a *single-command* program to run your model on the evaluation data, and produce plots/reports on its effectiveness.

Input: evaluation data with outcome variables.

Embedded: function coding current model

Output: summary statistics and distributions of predictions on data vs. outcome variables.

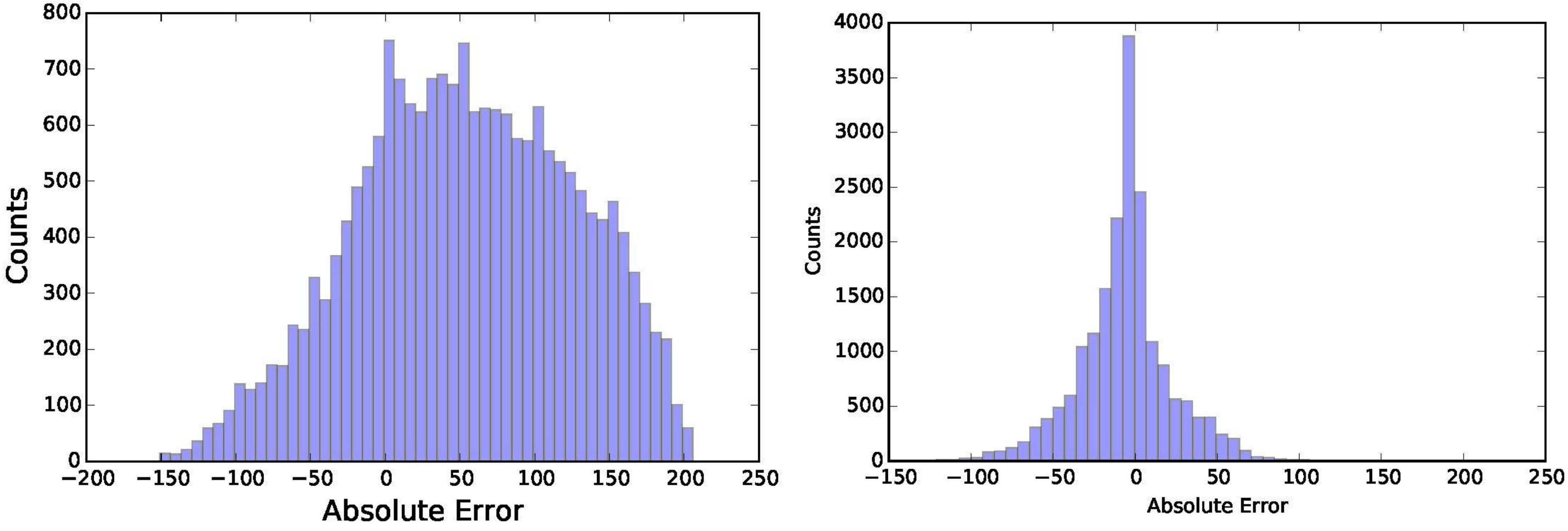
Evaluation Environment Architecture



Designing Good Evaluation Systems

- Produce error distributions in addition to binary outcomes (how close was your prediction, not just right or wrong).
- Produce a report with multiple plots / distributions automatically, to read carefully.
- Output relevant summary statistics about performance to quickly gauge quality.

Error Histograms: Dating Documents



Performance of Random vs. Naive Bayes models

Evaluation Environment: Results Table

	Dataset	Method	MAE	MedAE	Acc
0	NYTimes	Random	73.335463	65.0	0.004895
1	COHA_Fiction_100	Random	79.865017	72.0	0.005287
2	COHA_Fiction_500	Random	80.505849	74.0	0.003825
3	COHA_Fiction_1000	Random	80.604837	72.0	0.003825
4	COHA_Fiction_2000	Random	79.845332	72.0	0.005737
5	COHA_News_100	Random	66.539239	59.0	0.005461
6	COHA_News_500	Random	66.267091	59.0	0.005461
7	COHA_News_1000	Random	66.077670	57.5	0.004956
8	COHA_News_2000	Random	66.225526	58.0	0.005057

	Dataset	Method	MAE	MedAE	Acc
0	NYTimes	NB	21.306301	14	0.029728
1	COHA_Fiction_100	NB	32.302025	22	0.041732
2	COHA_Fiction_500	NB	25.428234	14	0.050056
3	COHA_Fiction_1000	NB	23.493926	13	0.053656
4	COHA_Fiction_2000	NB	22.493363	12	0.054781
5	COHA_News_100	NB	19.384001	14	0.030845
6	COHA_News_500	NB	16.657565	12	0.034891
7	COHA_News_1000	NB	16.282261	12	0.035093
8	COHA_News_2000	NB	16.220065	12	0.035599

Stratifying cases by topic and difficulty (length)

The Veil of Ignorance

A joke is not funny the second time because you already know the punchline.

Good performance on data you trained models on is very suspect, because models can easily be overfit.

Out of sample predictions are the key to being honest, if you have enough data/time for them.

Cross-Validation

Often we do not have enough data to separate training and evaluation data.

Train on *(k-1)/k* th of the data, evaluate on rest, then **repeat**, and average.

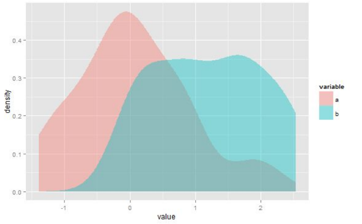
The win here is that you get a variance as to the accuracy of your model!

The limiting case is *leave one out validation.*

Amplifying Small Evaluation Sets

- **Create Negative Examples**: when positive examples are rare, all others are likely negative.
- **Perturb Real Examples**: This creates similar but synthetic ones by adding noise.
- **Give Partial Credit**: score by how far they are from the boundary, not just which side.

Probability Similarity Measures



There are several measures of distance between probability distributions

The **KL-Divergence** or information gain measures information lost replacing P with Q:

$$D_{\mathrm{KL}}(P\|Q) = \sum_i P(i) \ln \frac{P(i)}{Q(i)}.$$

Entropy is a measure of the information in a distribution.

Evaluation Statistics (Projects)

- Miss Universe?
- Movie gross?
- Baby weight?
- Art auction price?
- Snow on Christmas?
- Super Bowl / College Champion?
- Ghoul Pool?
- Future Gold / Oil Price?

Blackbox vs. Descriptive Models

Ideally models are descriptive, meaning they explain why they are making their decisions.

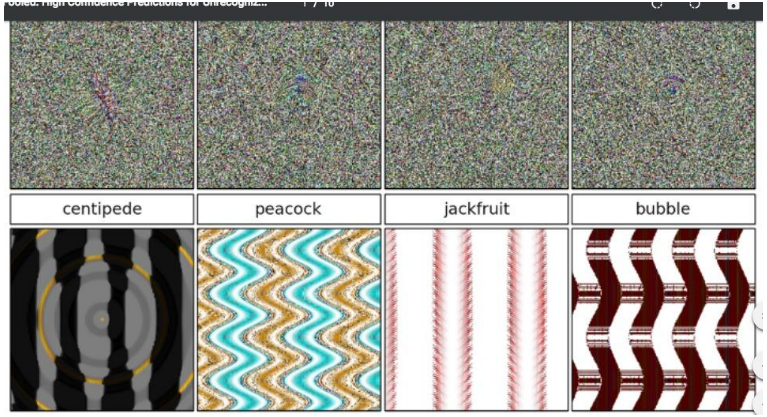
Linear regression models are descriptive, because one can see which variables are weighed heaviest.

Neural network models are generally opaque.

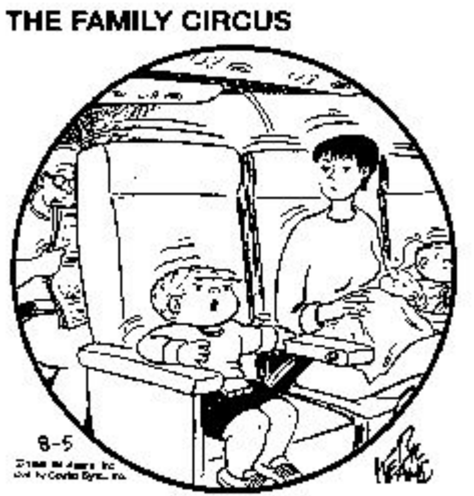
Lesson: “Distinguishing cars from trucks.”

Deep Learning Models are Blackbox

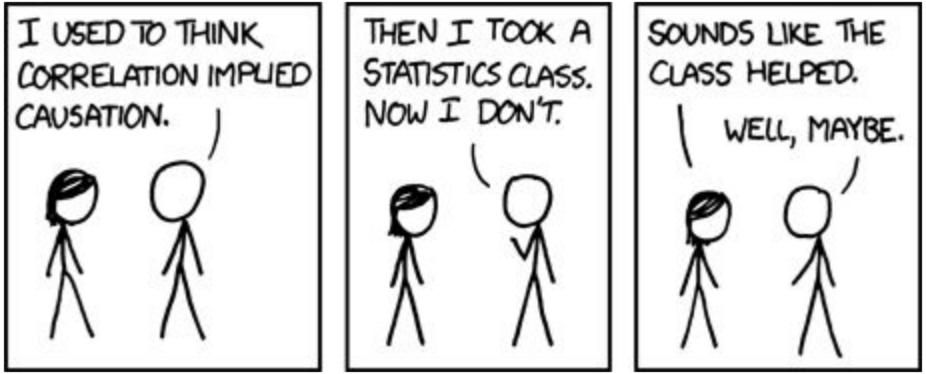
Deep learning models for computer vision are highly-effective, but opaque as to how they make decisions.
They can be badly fooled by images which would never confuse human observers.



Correlation Does Not Imply Causation



"I wish they didn't turn on that seatbelt sign so much! Every time they do, it gets bumpy."



Levels of Modeling

Interesting problems usually exist on several different levels, each of which require independent submodels.

Predicting the future price for a stock should involve submodels for analyzing (a) the general state of the economy, (b) its balance sheet, (c) the performance of its industrial sector, ...

Hierarchical Decomposition

Imposing a hierarchical structure on the model permits it to be built and evaluated in a logical and transparent way, instead of as a black box.

Often subproblems lend themselves to theory-based, first-principle models, which can then be used as features in a data-driven general model.

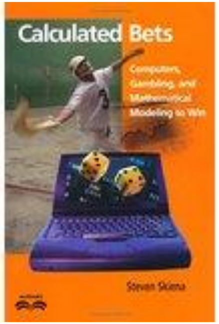
Simulation Models

"What I cannot create, I do not understand" (Feynman)

Monte Carlo simulation is the key to modeling systems of discrete events.

Our jai-alai betting system simulated games using

- Models of player skill
- Models of scoring system bias
- Models of bettor preferences



Levels of Modeling (Projects)

- Miss Universe?
- Movie gross?
- Baby weight?
- Art auction price?
- Snow on Christmas?
- Super Bowl / College Champion?
- Ghoul Pool?
- Future Gold / Oil Price?