# COMS 4721: Machine Learning for Data Science
## Lecture 1, 1/17/2017

Prof. John Paisley

Department of Electrical Engineering
& Data Science Institute

Columbia University

# Overview



This class will cover model-based techniques for extracting information from data with an end-task in mind. Such tasks include:

- predicting an unknown "output" given its corresponding "input"
- uncovering information within the data to better understand it
- data-driven recommendation, grouping, classification, ranking, etc.

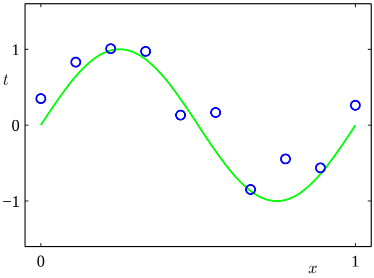
There are a few ways we can divide up the material as we go along, e.g.,

| | |
|---|---|
| supervised learning | unsupervised learning |
| probabilistic models | non-probabilistic models |
| modeling approach | optimization techniques |

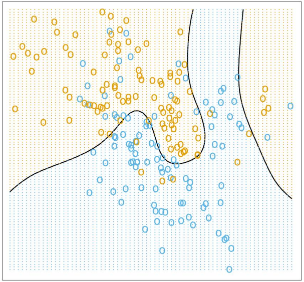
We'll adopt the first method and work in the second two along the way.

# Overview: Supervised Learning






(a) Regression



(b) Classification

**Regression**: Using set of inputs, predict real-valued output.

**Classification**: Using set of inputs, predict a discrete label (aka class).

# Example Classification Problem

Given a set of inputs characterizing an item, assign it a label.

## Is this spam?

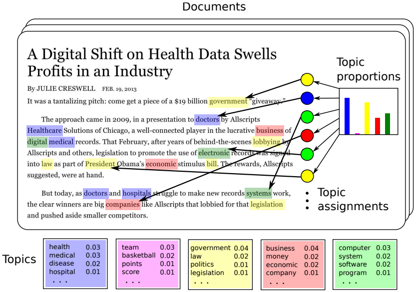
```
hi everyone,

i saw that close to my hotel there is a pub with bowling
(it's on market between 9th and 10th avenue).  meet
there at 8:30?
```

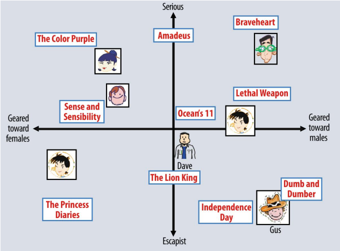
## What about this?

```
Enter for a chance to win a trip to Universal Orlando to
celebrate the arrival of Dr. Seuss's The Lorax on Movies
On Demand on August 21st!  Click here now!
```

# Overview: Unsupervised Learning




(c) topic modeling



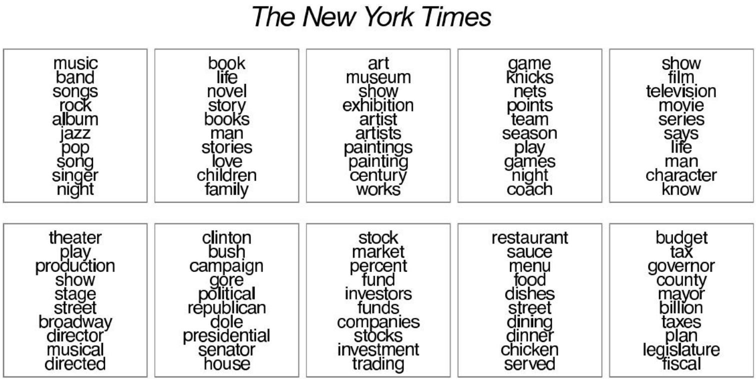

(d) recommendations[1]

With unsupervised learning our goal is often to uncover structure in the data. This helps with predictions, recommendations, efficient data exploration.

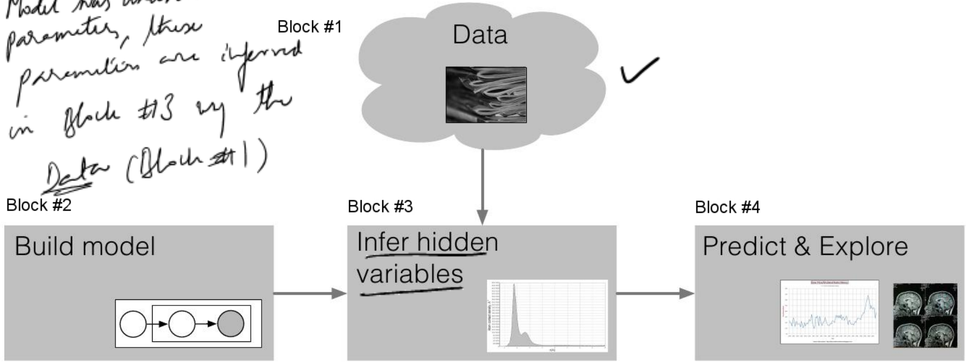
[1] Figure from Koren, Y., Robert B., and Volinsky, C.. "Matrix factorization techniques for recommender systems." Computer 42.8 (2009): 30-37.

# Example Unsupervised Problem

**Goal**: Learn the dominant topics from a set of news articles.



Model has unknown parameters, these parameters are inferred in Block #3 by the Data (Block #1)




Some algorithmic techniques are purely motivated by fact that we are doing a probabilistic model.

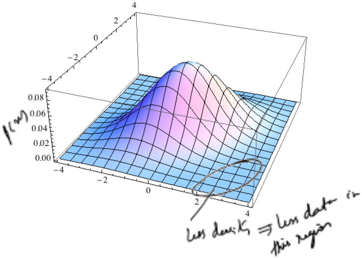
- Supervised vs. unsupervised: Blocks #1 and #4
- Probabilistic vs. non-probabilistic: Primarily Block #2 (Some Block #3)

However a lot of techniques for non-prob models apply to prob model.

- Model development (Block #2) vs. Optimization techniques (Block #3)

## Gaussian density in $d$ dimensions

- Block #1: Data $x_1, \ldots, x_n$. Each $x_i \in \mathbb{R}^d$
- Block #2: An i.i.d. Gaussian model
- Block #3: Maximum likelihood
- Block #4: Leave undefined




The density function is

$$p(x|\mu, \Sigma) := \frac{1}{(2\pi)^{\frac{d}{2}}\sqrt{\det(\Sigma)}} \exp\Big(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\Big)$$

The central moments are:

$$\mathbb{E}[x] = \int_{\mathbb{R}^d} x\, p(x|\mu, \Sigma)dx = \mu,$$

$$\mathrm{Cov}(x) = \mathbb{E}[(x - \mathbb{E}[x])(x - \mathbb{E}[x])^T] = \mathbb{E}[xx^T] - \mathbb{E}[x]\mathbb{E}[x]^T = \Sigma.$$

## Probabilistic Models

- A *probabilistic model* is a set of probability distributions, $p(x|\theta)$.
- We pick the *distribution family* $p(\cdot)$, but don't know the parameter $\theta$.

**Example**: Model data with a Gaussian distribution $p(x|\theta)$, $\theta = \{\mu, \Sigma\}$.

$\mu, \Sigma$ are inferred in Block #3

## The i.i.d. assumption

Assume data is *independent and identically distributed (iid)*. This is written

$$x_i \overset{iid}{\sim} p(x|\theta), \quad i = 1, \ldots, n.$$

training data is produced independent of other samples

Writing the density as $p(x|\theta)$, then the *joint* density decomposes as

$$p(x_1, \ldots, x_n|\theta) = \prod_{i=1}^{n} p(x_i|\theta).$$

↳ Objective function = f (data & unknown parameter)

## Maximum Likelihood approach

We now need to find $\theta$. *Maximum likelihood* seeks the value of $\theta$ that maximizes the likelihood function:

$$\boxed{\hat{\theta}_{\text{ML}} := \arg\max_{\theta} \; p(x_1, \ldots, x_n | \theta),}$$

→ ⓐ

This value best explains the data according to the chosen distribution family.

## Maximum Likelihood equation

The analytic criterion for this maximum likelihood estimator is:

$\boxed{\nabla_\theta f(x_i|\theta) = 0}$

$$\nabla_\theta \prod_{i=1}^{n} p(x_i|\theta) = 0.$$

→ ⓐ

Max m of a function (i.e ⓐ) is where gradient = 0.

Simply put, the maximum is at a peak. There is no "upward" direction.

## Logarithm trick

Calculating $\nabla_\theta \prod_{i=1}^{n} p(x_i|\theta)$ can be complicated. We use the fact that the logarithm is monotonically increasing on $\mathbb{R}_+$, and the equality

$$\ln\Big(\prod_i f_i\Big) = \sum_i \ln(f_i).$$

Consequence: Taking the logarithm does not change the *location* of a maximum or minimum:

$$\max_y \ \ln g(y) \neq \max_y \ g(y) \qquad \text{The } \textit{value} \text{ changes.}$$

$$\boxed{\arg\max_y \ \ln g(y) = \arg\max_y \ g(y)} \qquad \text{The } \textit{location} \text{ does not change.}$$

(true for positive function $g(y)$)

## Maximum likelihood and the logarithm trick

$$\hat{\theta}_{\text{ML}} = \arg\max_{\theta} \prod_{i=1}^{n} p(x_i|\theta) = \arg\max_{\theta} \ln\left(\prod_{i=1}^{n} p(x_i|\theta)\right) = \arg\max_{\theta} \sum_{i=1}^{n} \ln p(x_i|\theta)$$

To then solve for $\hat{\theta}_{\text{ML}}$, find

$$\nabla_\theta \sum_{i=1}^{n} \ln p(x_i|\theta) = \sum_{i=1}^{n} \nabla_\theta \ln p(x_i|\theta) = 0.$$

Depending on the choice of the model, we will be able to solve this

1. analytically (via a simple set of equations)
2. numerically (via an iterative algorithm using different equations)
3. approximately (typically when #2 converges to a local optimal solution)

# Example: Multivariate Gaussian MLE



→ covariance matrix is always +'ve semi-definite.

## Block #2: Multivariate Gaussian data model

Model: Set of all Gaussians on $\mathbb{R}^d$ with unknown mean $\mu \in \mathbb{R}^d$ and covariance $\Sigma \in \mathbb{S}^d_{++}$ (positive definite $d \times d$ matrix).

We assume that $x_1, \ldots, x_n$ are i.i.d. $p(x|\mu, \Sigma)$, written $x_i \overset{iid}{\sim} p(x|\mu, \Sigma)$.

## Block #3: Maximum likelihood solution

We have to solve the equation

$$\sum_{i=1}^{n} \nabla_{(\mu,\Sigma)} \ln p(x_i|\mu, \Sigma) = 0$$

for $\mu$ and $\Sigma$. (Try doing this without the log to appreciate it's usefulness.)

# Example: Gaussian Mean MLE



First take the gradient with respect to $\mu$.

$$0 = \nabla_\mu \sum_{i=1}^{n} \ln \frac{1}{\sqrt{(2\pi)^d|\Sigma|}} \exp\Big(-\frac{1}{2}(x_i-\mu)^T\Sigma^{-1}(x_i-\mu)\Big)$$

$$= \nabla_\mu \sum_{i=1}^{n} -\frac{1}{2}\ln(2\pi)^d|\Sigma| - \frac{1}{2}(x_i-\mu)^T\Sigma^{-1}(x_i-\mu)$$

$$= -\frac{1}{2}\sum_{i=1}^{n} \nabla_\mu \Big(x_i^T\Sigma^{-1}x_i - 2\mu^T\Sigma^{-1}x_i + \mu^T\Sigma^{-1}\mu\Big) = -\Sigma^{-1}\sum_{i=1}^{n}(x_i-\mu)$$

Since $\Sigma$ is positive definite, the only solution is

Generally $\Sigma$ covariance matrices are +ve semi-definite i.e $\geq 0$

$$\sum_{i=1}^{n}(x_i-\mu) = 0 \qquad \Rightarrow \qquad \hat{\mu}_{\text{ML}} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Since this solution is independent of $\Sigma$, it doesn't depend on $\hat{\Sigma}_{\text{ML}}$.

# Example: Gaussian Covariance MLE



Now take the gradient with respect to $\Sigma$.

$$
\begin{aligned}
0 &= \nabla_\Sigma \sum_{i=1}^{n} -\frac{1}{2}\ln(2\pi)^d|\Sigma| - \frac{1}{2}(x_i-\mu)^T\Sigma^{-1}(x_i-\mu) \\
&= -\frac{n}{2}\nabla_\Sigma \ln|\Sigma| - \frac{1}{2}\nabla_\Sigma \mathrm{trace}\left(\Sigma^{-1}\sum_{i=1}^{n}(x_i-\mu)(x_i-\mu)^T\right) \\
&= -\frac{n}{2}\Sigma^{-1} + \frac{1}{2}\Sigma^{-2}\sum_{i=1}^{n}(x_i-\mu)(x_i-\mu)^T
\end{aligned}
$$

Solving for $\Sigma$ and plugging in $\mu = \hat{\mu}_{\mathrm{ML}}$,

$$
\hat{\Sigma}_{\mathrm{ML}} = \frac{1}{n}\sum_{i=1}^{n}(x_i-\hat{\mu}_{\mathrm{ML}})(x_i-\hat{\mu}_{\mathrm{ML}})^T.
$$

So if we have data $x_1, \ldots, x_n$ in $\mathbb{R}^d$ that we hypothesize is i.i.d. Gaussian, the maximum likelihood values of the mean and covariance matrix are

$$\hat{\mu}_{\text{ML}} = \frac{1}{n}\sum_{i=1}^{n} x_i, \quad \hat{\Sigma}_{\text{ML}} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu}_{\text{ML}})(x_i - \hat{\mu}_{\text{ML}})^T.$$

**Are we done?** There are many assumptions/issues with this approach that makes finding the "best" parameter values not a complete victory.

- We made a model assumption (multivariate Gaussian). Block #2
- We made an i.i.d. assumption.
- We assumed that maximizing the likelihood is the best thing to do.

Comment: We often use $\theta_{\text{ML}}$ to make predictions about $x_{new}$ (Block #4).
How does $\theta_{\text{ML}}$ generalize to $x_{new}$?
If $x_{1:n}$ don't "capture the space" well, $\theta_{\text{ML}}$ can *overfit* the data. - MLE fails here.

In block #3, we chose an objective function (i.e. likelihood) to model the data. Later we maximized it cause we chosen likelihood. Choosing likelihood ~~thing~~ was an assumption that it is best thing to do.