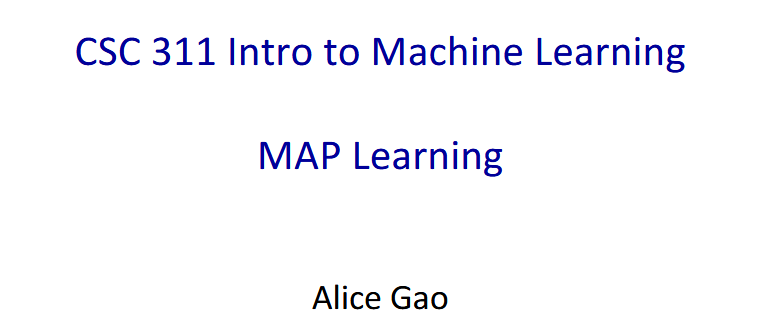
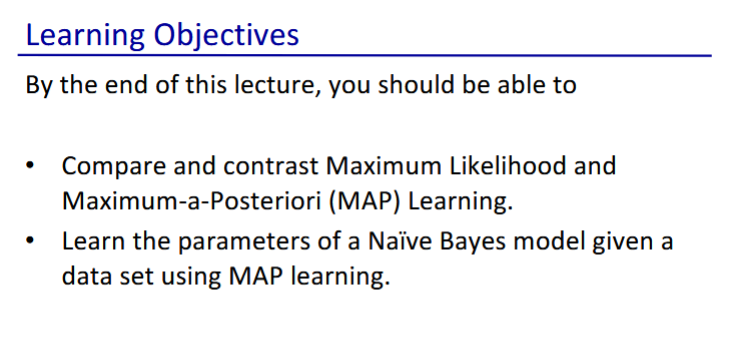
|  |
| --- |
| **Admin stuff**   * No strike occurring * Test 2 on friday   + Vote on review session time on piazza     - What to put on aid sheet   + 1-sided aid sheet, can bring a calculator (but likely won’t be needed)   + Neural networks up to 2 layers (1 hidden, 1 output)   + Weeks 4-7     - Also some essential introductory materials included (overfitting, underfitting, test/validation set, etc.) * A2 solutions posted tomorrow * **Research projects on usra.cs.toronto.edu**   **Disadvantages of maximal likelihood (MLE)**   * Maximal likelihood can overfit when the number of trials are low * Example: if we flip a coin twice and get only heads, an MLE model will assign 100% to heads and never predict tails   **Maximum-A-Posteriori (MAP) learning**   * MAP seeks parameter () values that maximise the posterior probability of parameter values given the data () * MAP is essentially a more general form of MLE that incorporates a prior assumption about the parameter   + MLE is MAP but with a uniform prior   + MAP     - is the prior - starting assumption about the parameter, and is a hyperparameter we provide   + MLE     - * is the uniform probability distribution, we can add this since it does not change which gives the max * **Conflict between prior and data**   + When there is little training data, MAP uses the priors to help make the model   + When there is a lot of training data, MAP can disregard the prior and use the training data to make the model * **Log-likelihood derivation for coin flip**   + - Parameters a and b of the beta distribution of prior essentially act as added trials to our data       * See next note about beta distribution   + Maths on slide 13   **Beta distribution**   * A type of equation that is able to produce a wide variety of probability distributions   + 2 parameters: **a** and **b** that influence the shape of the distribution     - When a is larger, we are biassed towards a larger theta     - When b is larger, we are biassed against a smaller theta     - corresponds to the uniform distribution   + Note: probability is proportional, as there is an omitted constant term (no effect on max ) * The beta distribution is useful for defining our prior distribution for MAP learning   **Gaussian discriminant analysis (GDA)**   * Model that fits a Gaussian distribution for each output class to make predictions   + Allows for continuous features, unlike Naive Bayes which can only do discrete features   + Gaussian distribution also called normal distribution * **Process**   + We first fit a Gaussian distribution for each class ()     - Results in a multivariate Gaussian distribution if we have multiple features   + We then make a prediction using Bayes theorem * **Gaussian distribution**   + Each Gaussian distribution is defined by a mean vector () (D x 1) and a covariance matrix () (D x D)      - Mean vector defines where the distribution is centred     - Covariance matrix defines how the distribution behaves around the mean   + The ideal mean and covariance for the model is the mean and covariance of the training data * **Probability distribution function (PDF) of Gaussian distribution on slide 16** |

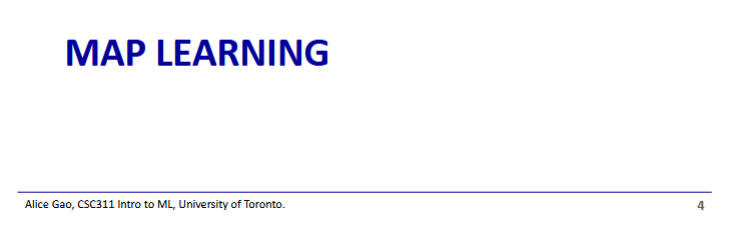
Recap: Naive Bayes

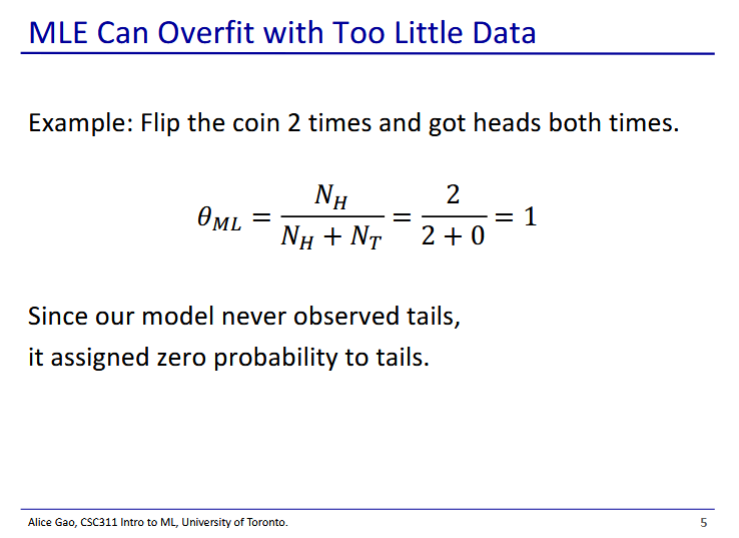
* Model makes a simplifying assumption that all features are conditionally independent given the class label
  + Benefit is to allow us to decompose the joint distribution into individual independent probabilities
* Maximum likelihood
  + We look at the conditional probability of each feature, and decide on the most likely class based on what features are present



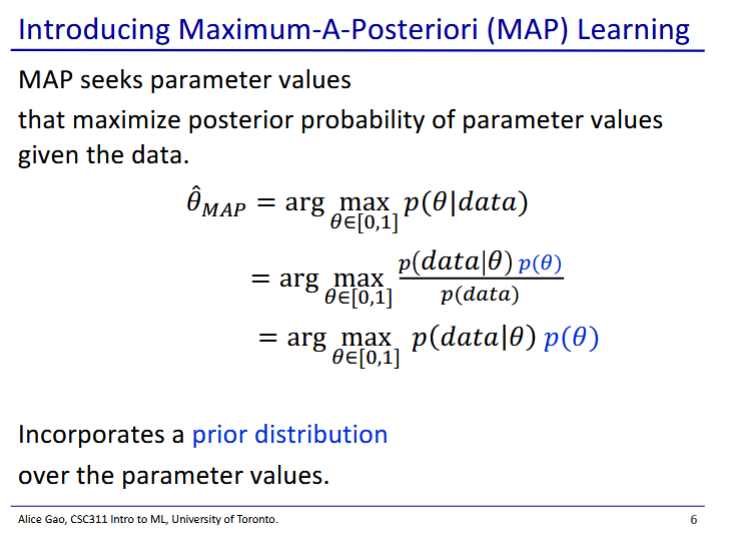




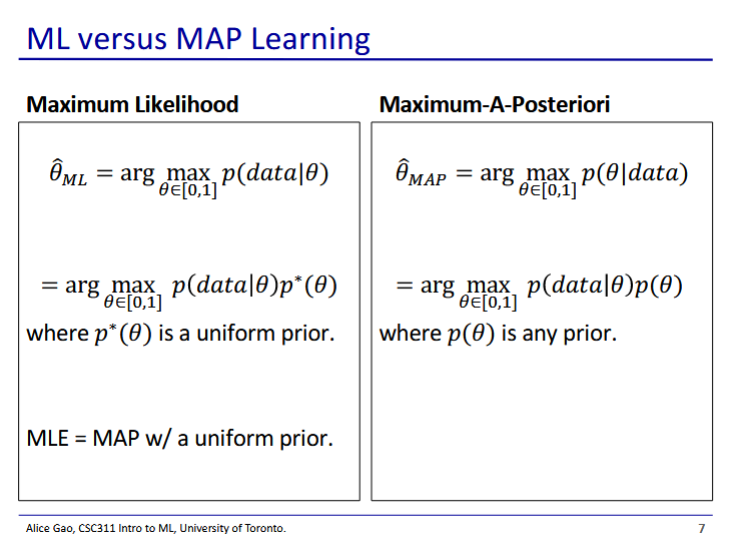




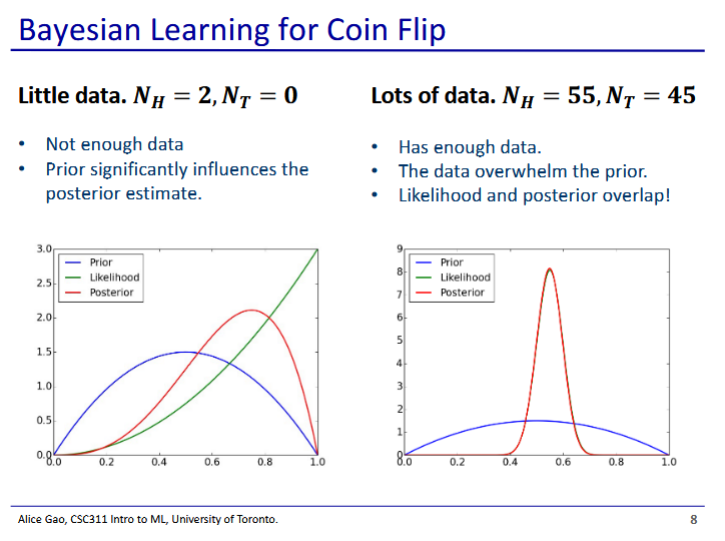
* Maximal likelihood (from last week) can overfit when we have too little data
* Since our training data had no tails, our model never assigns any probability to tails
  + This is a very extreme prediction that we don’t want



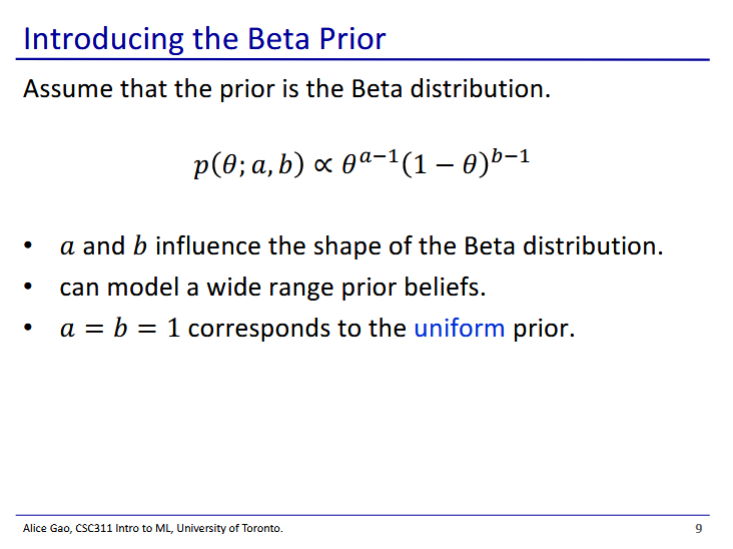
* MAP find a parameter value that maximises the posterior probability of the model
  + Posterior probability =
* We maximise the posterior probability using bayes rule
* We start with some assumption about the probability on the coin, then look at how likely the training data is based on a predicted probability on the coin, taking into account our assumption
  + We look for the bias on the coin that gives the maximal probability of getting our training data
* From lines 2-3 we get rid of the denominator (as it is a constant that doesn’t matter when we are trying to maximise)



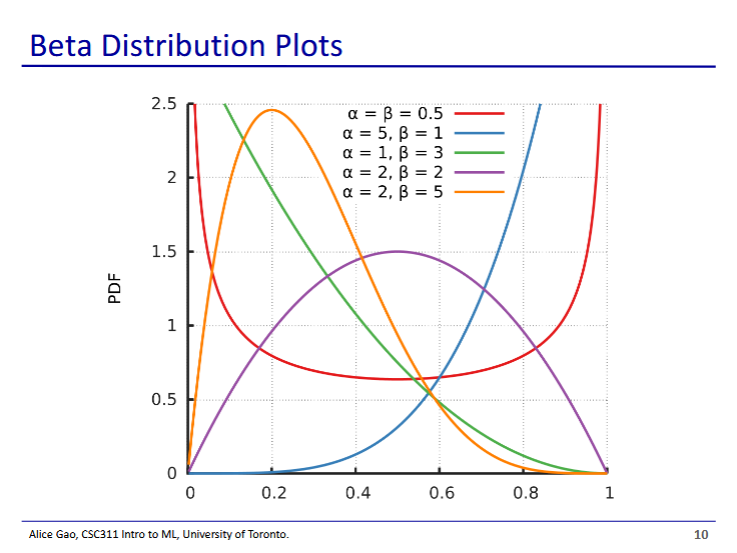
* We can write maximum likelihood as MAP but with a uniform prior ( has an equal probability to be any value)
  + It is ok to add the extra term since it is a uniform distribution, and thus is constant for every
    - Thus does not affect which gives the largest value of
* MAP is a general version of MLE
  + Works for every probability distribution of



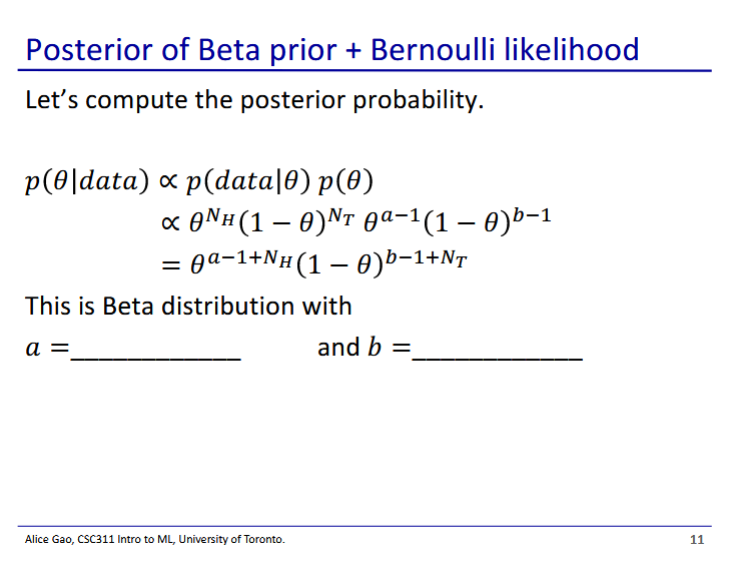
* Once we have added a prior, there is a struggle between whether the model trusts the prior more or the data
  + Prior is a belief that the model has about the distribution before it even sees the data
* Left: Prior (blue) and a little bit of data (likelihood) (green)
  + Posterior looks like a compromise between the prior and the data
  + The prior has a stronger effect on the posterior when we have less data
* Right: Prior and a lot of data (green hidden behind the red)
  + Posterior matches with the data
  + We have enough data to disregard our priors (we trust the data more)



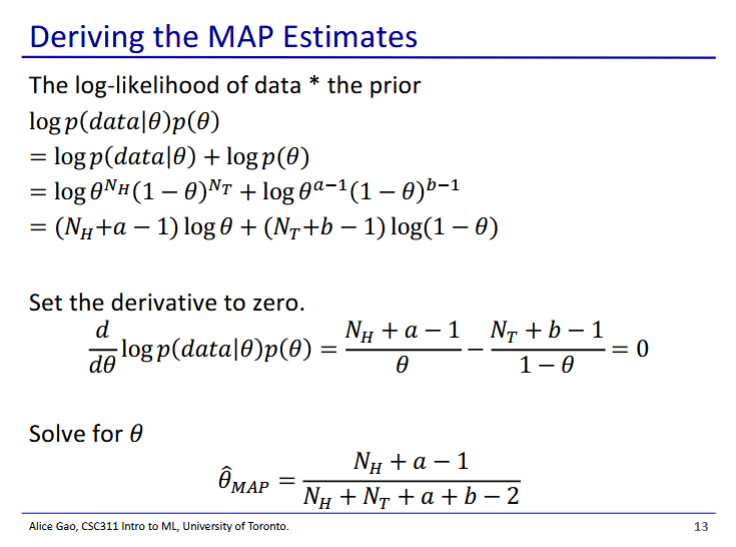
* Beta distribution
  + 2 parameters: a and b (or alpha and beta)
  + Picking values for a and b gives a probability distribution for
* Note: this slide has proportional, since there is an omitted constant factor (constant factor will not affect our optimisation)



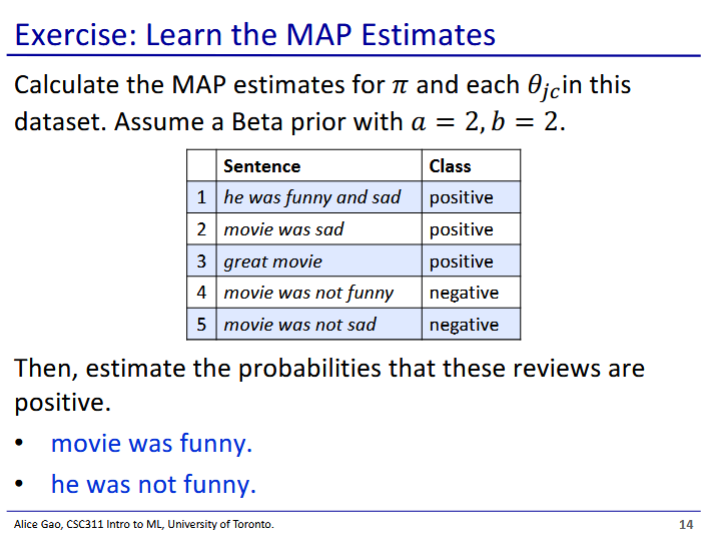
* Depending on what we choose for a and b, our beta distribution will look very different
  + X axis is value of
* If we have both a and b equal to 1, we get a uniform distribution
* When a is larger, we are biased towards a larger theta
* When b is larger, we are biased against a smaller theta



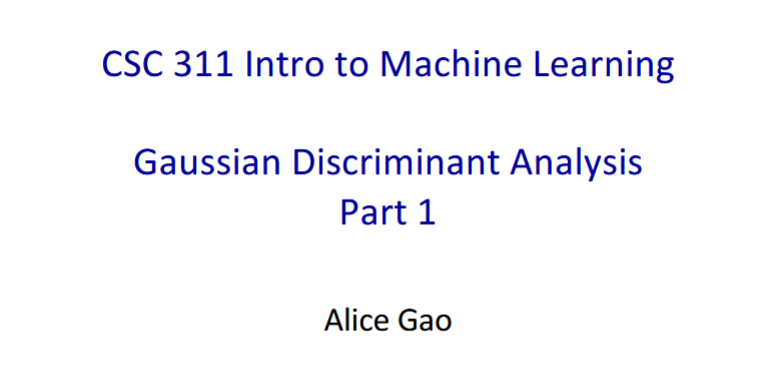
* We can interpret the result as a beta distribution
  + New alpha is a+NH
  + New beta is b+HT
* Note that the prior and the likelihood look very similar
* Also that the prior and the posterior look very similar
* We can interpret a and b as fake trials that we run before our actual trials that give us a preexisting idea of the distribution
* Beta prior + bernouli likelihood -> beta posterior
  + Beta is a conjugate prior for the bernoulli likelihood

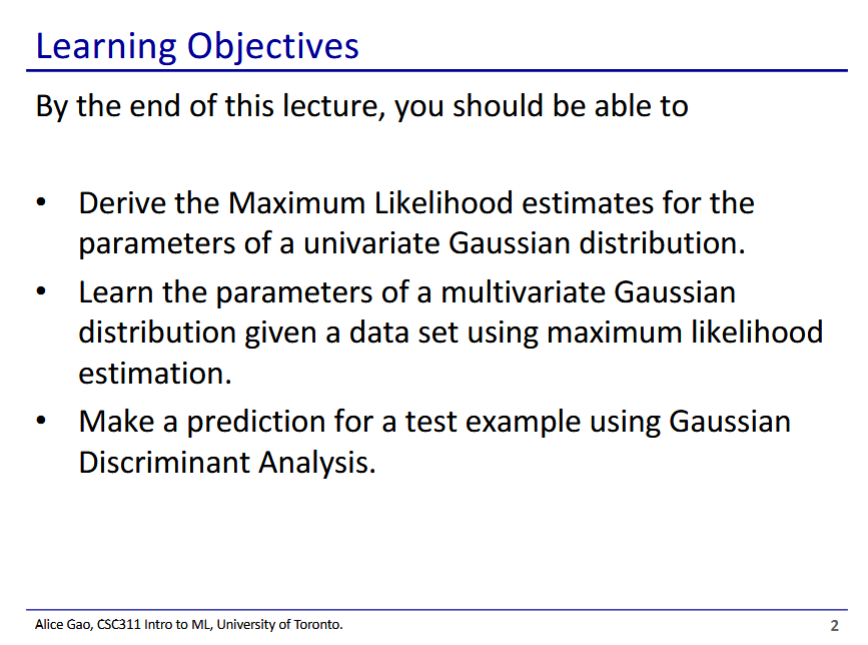


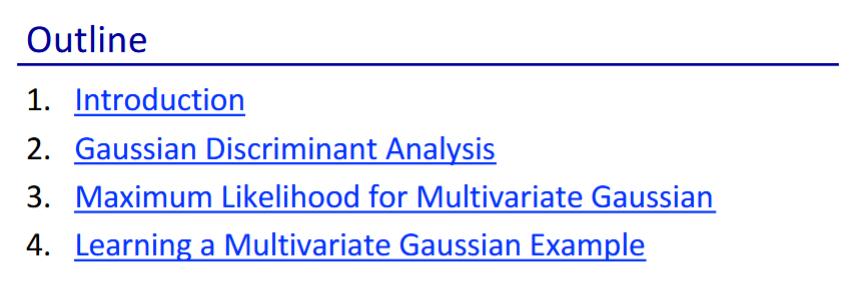
* Almost identical for the derivation for log-likelihood, except we multiply by the prior
* Recall: for log-likelihood without priors:
  + The beta distribution functions essentially as added trials to our data



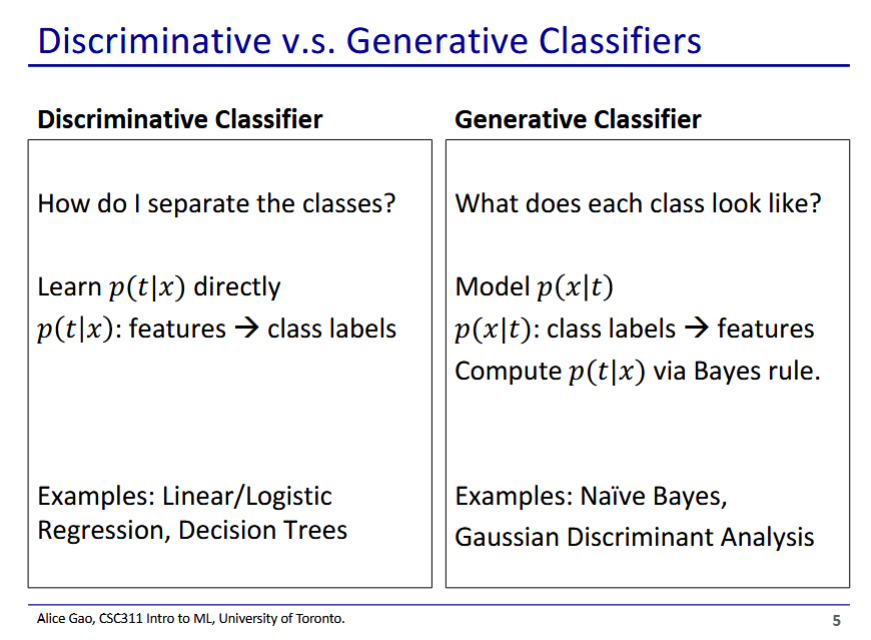
* Answers for predictions are available on the hidden slides

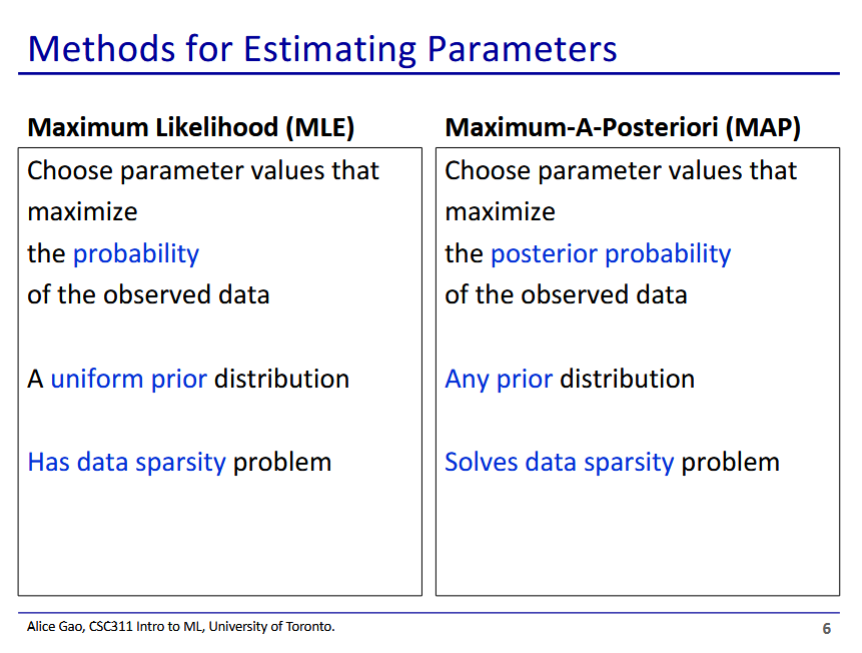


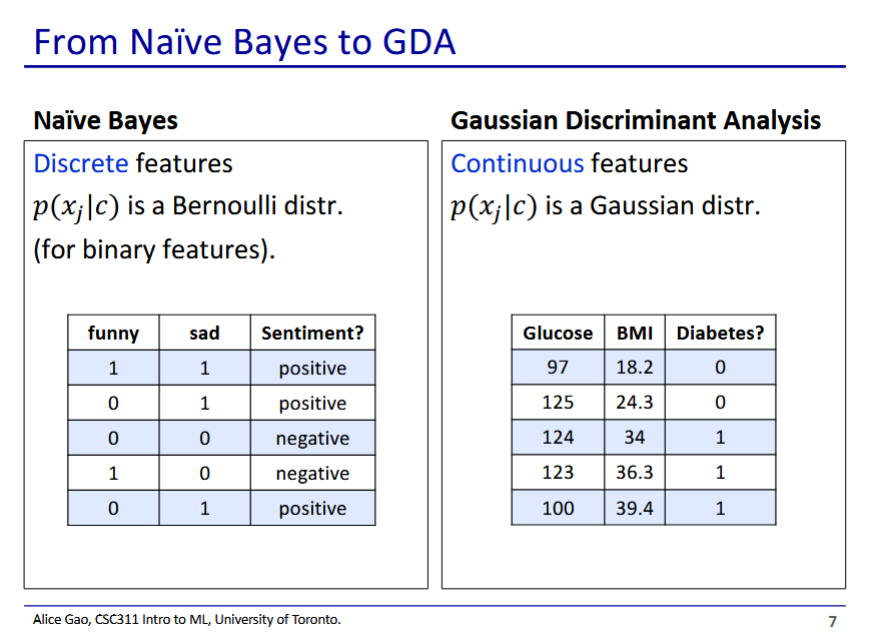






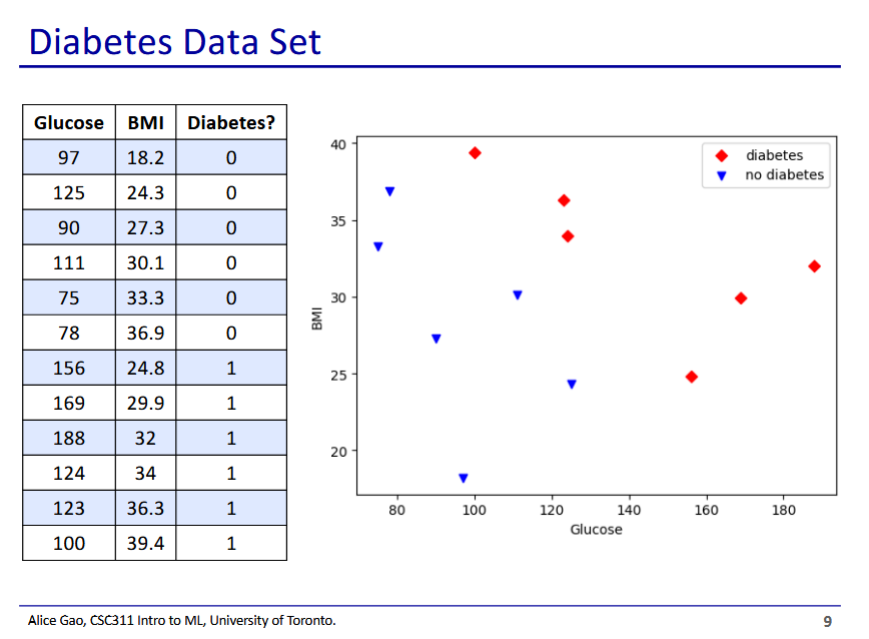




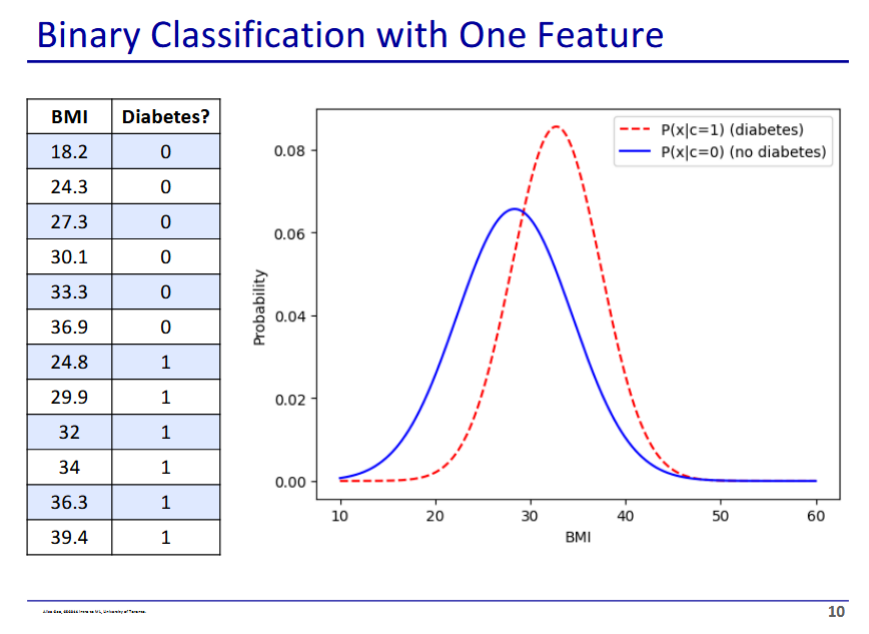


* Lets us map continuous features
* Naive Bayes MLE and MAP are limited to discrete features

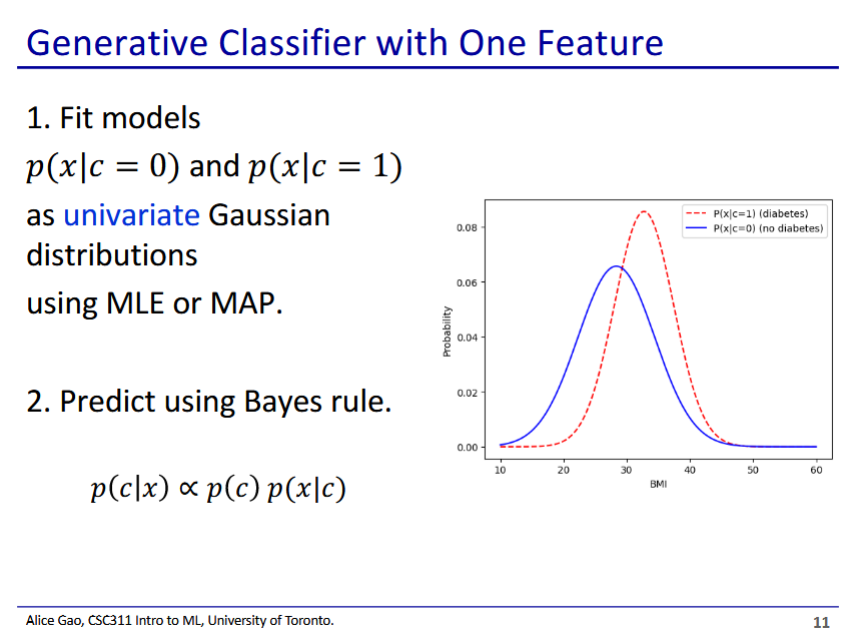




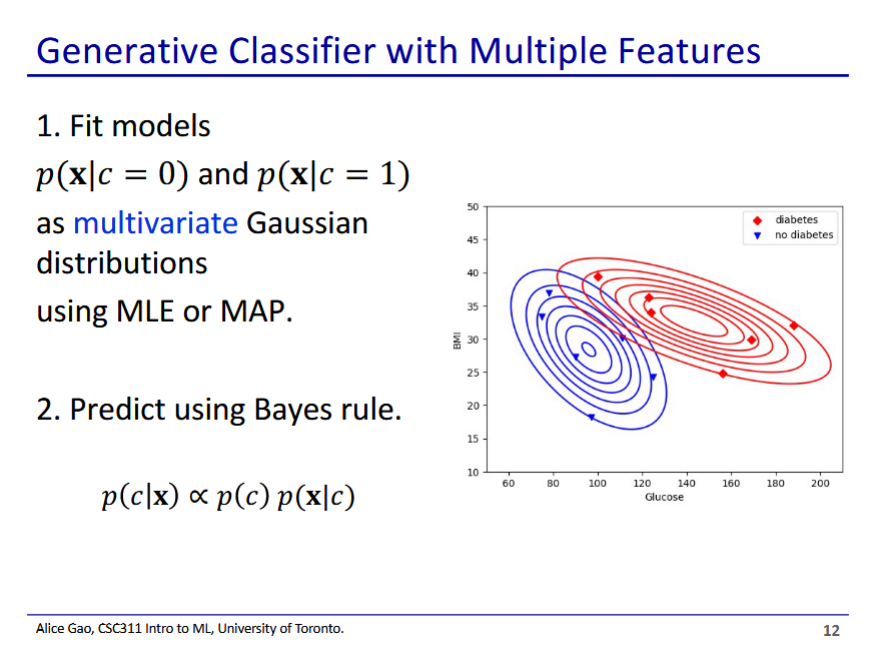
* Diabetes dataset used as the example
* Contains 2 continuous features: glucose and BMI



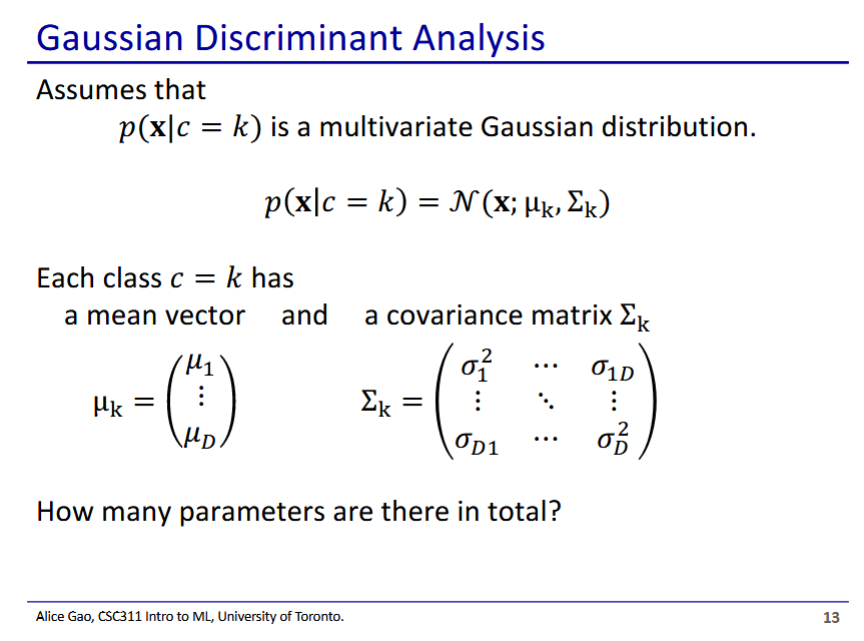
* In this case we only look at 1 feature: BMI
* We get a univariate Gaussian distribution
  + Gaussian distribution is the same as normal distribution



* We split data into 2 cases along the class
* We then learn a gaussian distribution for each class
* Then we predict using bayes rule
  + Same equation as with naive Bayes

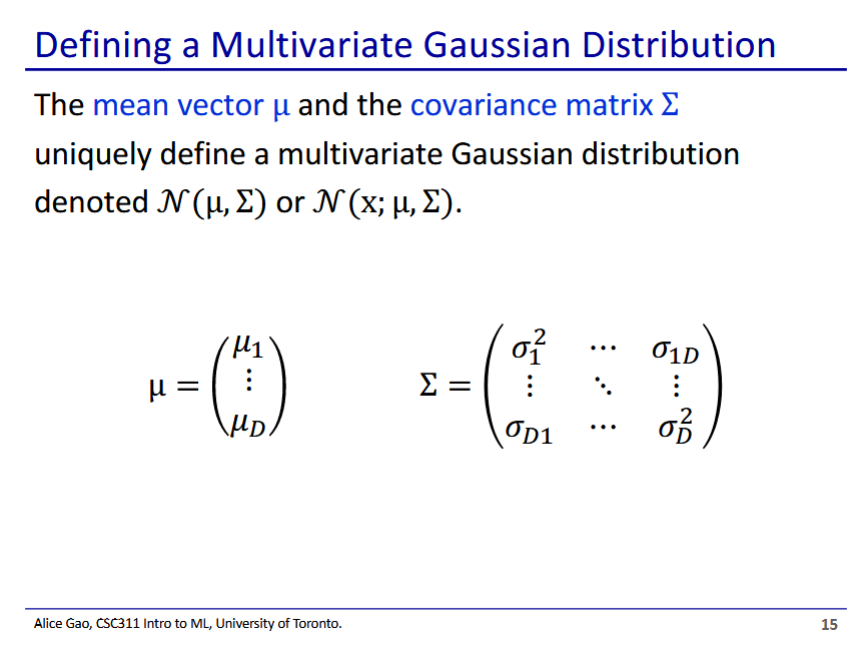


* Process is still the same, except now we fit the data into a multivariate Gaussian distribution
  + Probability graph with 2 features is in 3D

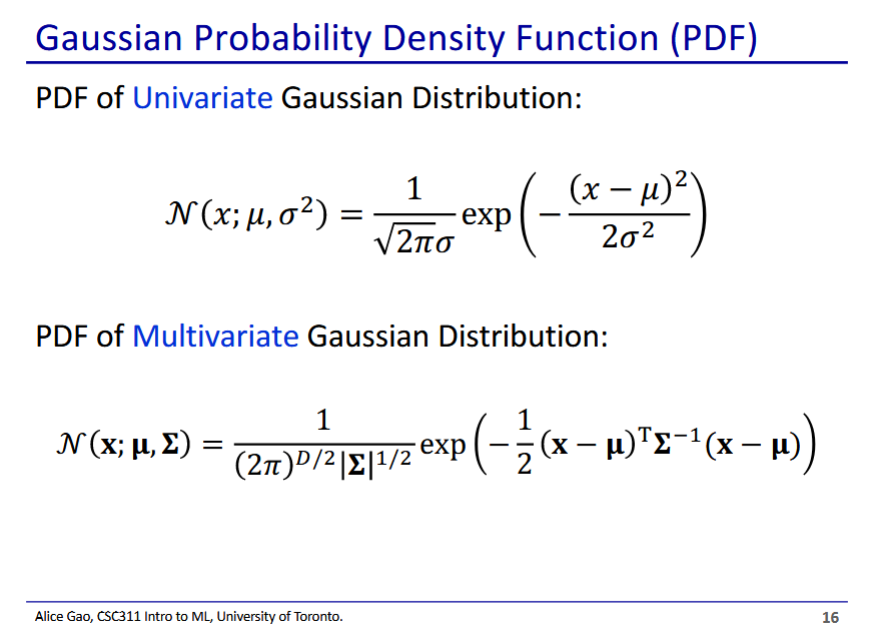


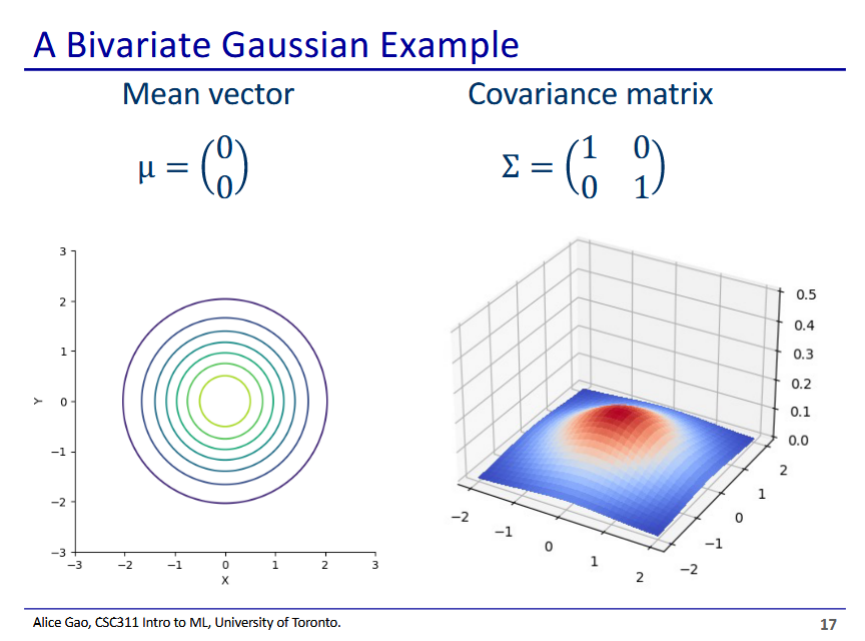
* Given a class, probability of a set of given feature values follows gaussian distribution
  + Fancy N means gaussian distribution
* To pin down the distribution for each class we need a mean vector and a covariance matrix
  + We have k\*(D^2 + D) parameters
  + D^2 + D parameters for each class



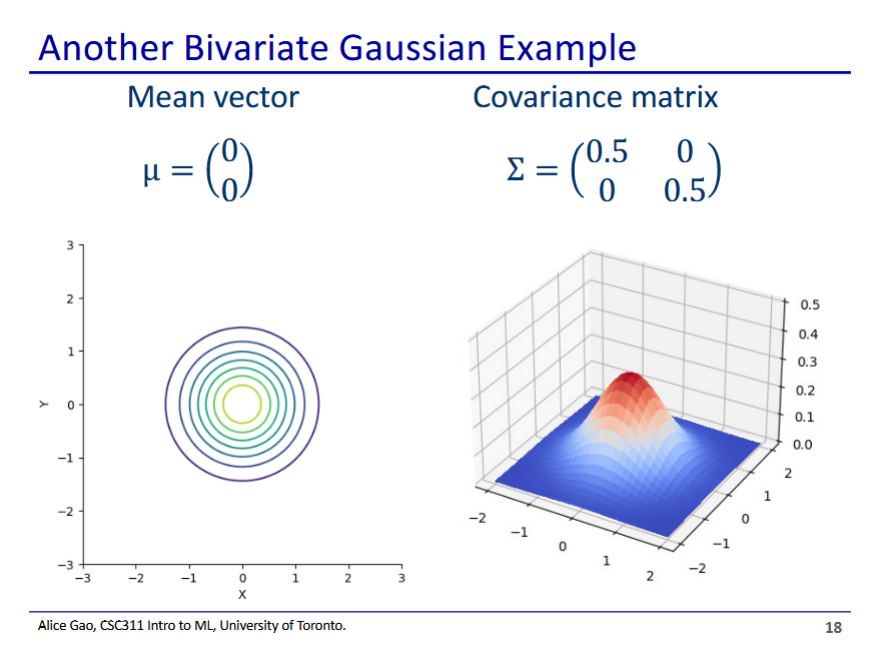


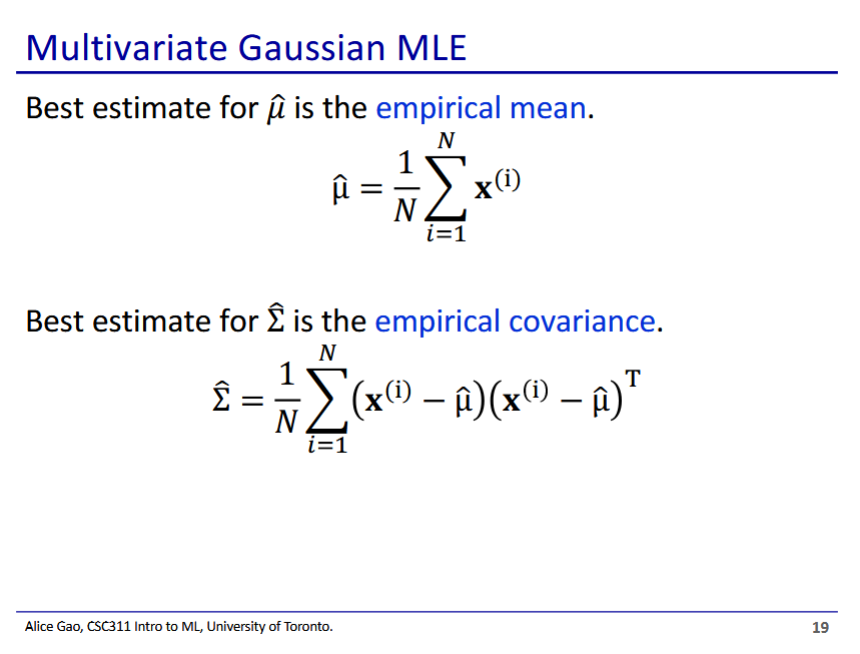
* Gaussian distribution is defined using 2 parameters: a mean vector and a convergence matrix
* Mean vector defines where distribution is centred
* Convergence matrix defines how distribution behaves around the mean





* 2-feature distribution with mean of zero vector

****



* Estimate mean vector and covariance matrix using maximum likelihood
  + This is a similar principle to naive bayes MLE
* Best covariance is the covariance of the data, best mean is the mean of the data