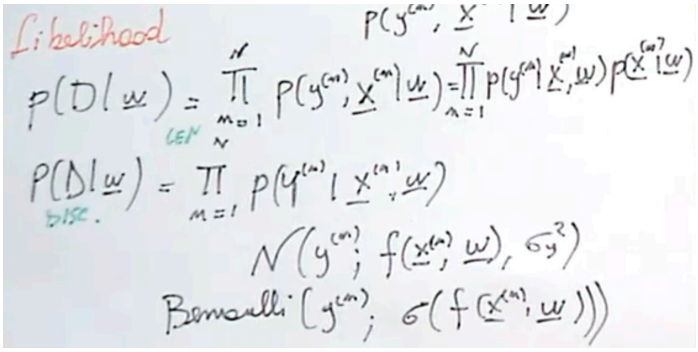
Week 10a Bayesian logistic regression

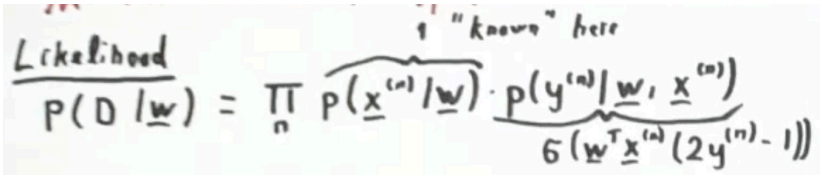
Recap:
Task: given data D = {x,y}, fit probabilistic model
Discriminative model, y given x P(y | x, w)
Generative model, modeling x and y directly P(y, x | w)

Then specify likelihood = P (D | w) = P (x_train, y_train | w)
Discriminative model: product over each datapoints m, then product over sigmoid, z = [-1,1] = 2y-1 (or other functions)
Generative model: product over each datapoints m, then product rule

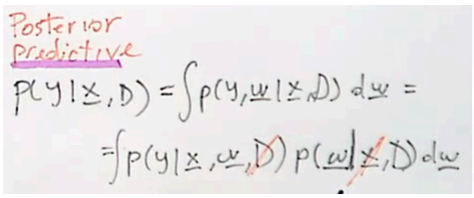


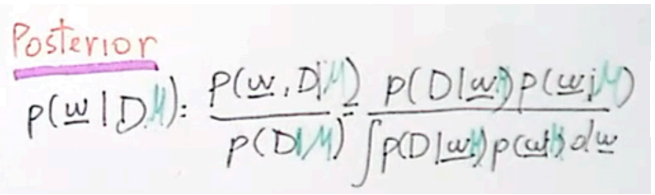
For discriminative logistic regression: why product over P(y^(m) | w, x^(m))
minimize negative log likelihood, find single w*

Likelihood is a function of parameters, not a distribution over the parameters
likelihood is product over the probabilities according to the model of observing the datapoints
Assume data is identically distributed and independently drawn from this distribution (IID) (so get the normal distribution symbol)

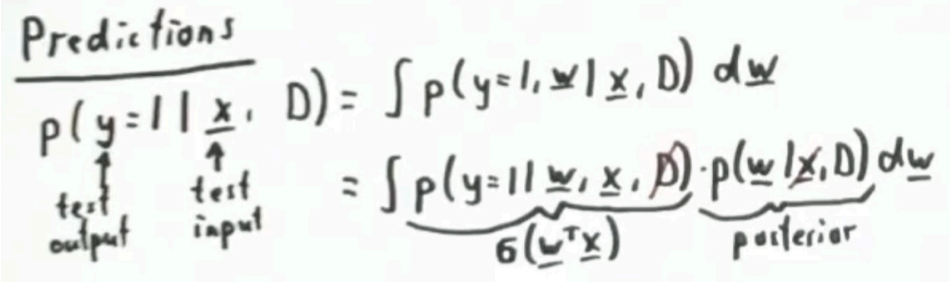


Bayesian approach: true set of weights are unknown, described by prob dist
predictions P(y_test | x_test, D) are made considering all possible settings (sigmoids), weighted by how plausible they are given the training data (posterior p(w|D))





$$P(\mathcal{D}) = \int P(\mathcal{D} \mid \mathbf{w})\, p(\mathbf{w})\, \mathrm{d}\mathbf{w}.$$

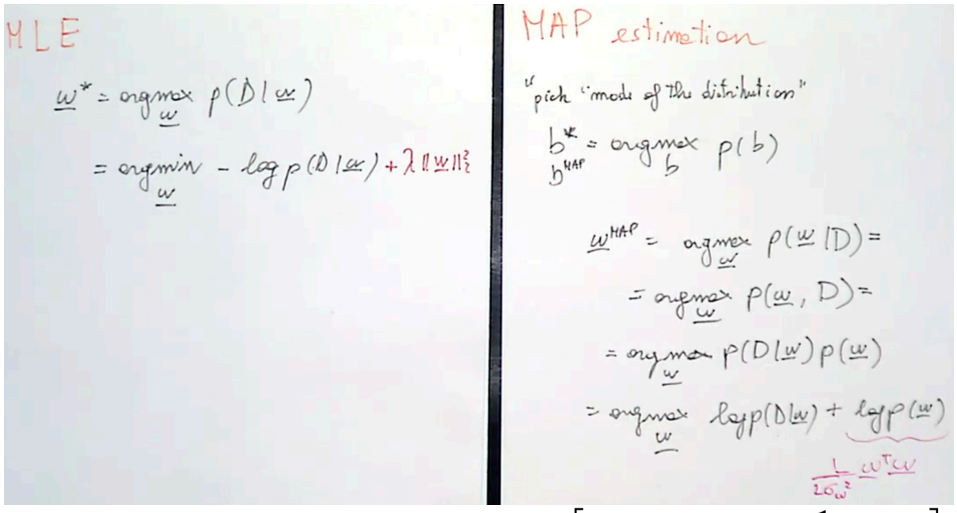


Interpretation: average the predictive distributions P(y | x, w) for different parameters weighted by how plausible those parameters are, p(w | D)

How to compute this integral?
(= expectation of sigmoid function P(ytest | xtest, w) over posterior)

if prior * likelihood yields a parametric form that we know of (normal), can calculate integral in closed form
or approximate it by Gaussian (simpler form), optimize a “simpler surrogate”
or match moments (first moment = mean & second moment = variance)
or MAP (= normal logistic regression)

MAP (maximum probability inference):
Get point estimate
= just pick mode of the distribution (pdf curve highest)
= argmax of posterior P(w|D), (= get most probable weight in posterior)
= argmax of P(w,D), since P(D) does not depend on parameters w
then chain rule, take log
=> Under certain assumptions (prior is simple independent gaussian, zero mean & scalar variance), prior = regularization term
=> MAP = L2 regularized MLE (logistic regression)



$$\mathbf{w}^* = \arg\max_{\mathbf{w}} [\log p(\mathbf{w} \mid \mathcal{D})] = \arg\max_{\mathbf{w}} \left[\log P(\mathcal{D} \mid \mathbf{w}) - \frac{1}{2\sigma_w^2}\mathbf{w}^\top \mathbf{w}\right]$$

Advantages of bayesian:

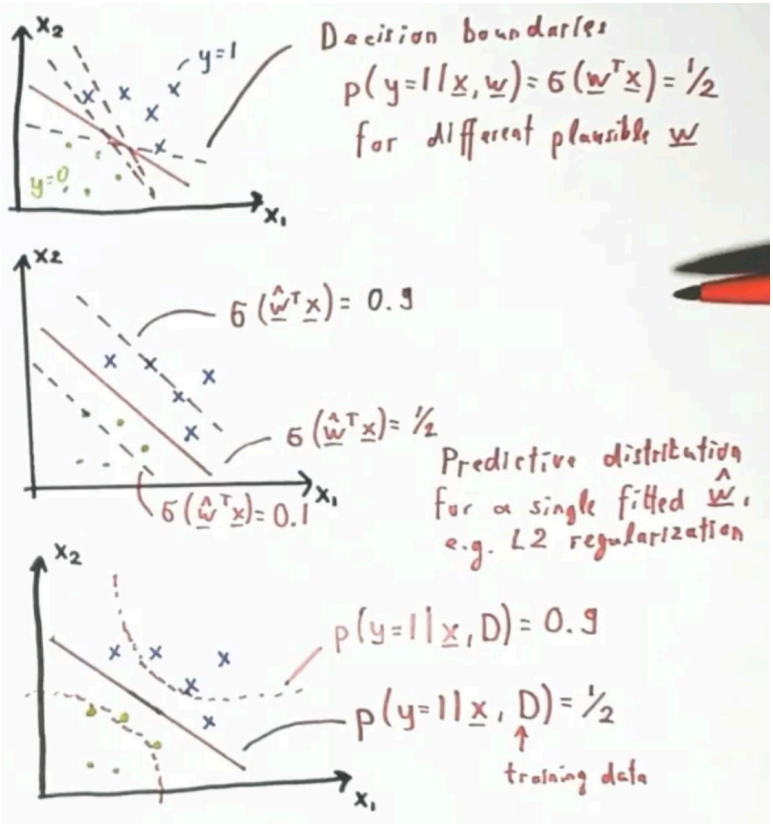
model uncertainty

look at marginal likelihood, make better use of available data (to choose model, also don't need validation set)

posterior represent our belief about the model

Represent uncertainty (in first figure)

Last figure: when further away from the training inputs, Contours curved, predictions less certain



Sketch logistic regression posterior:

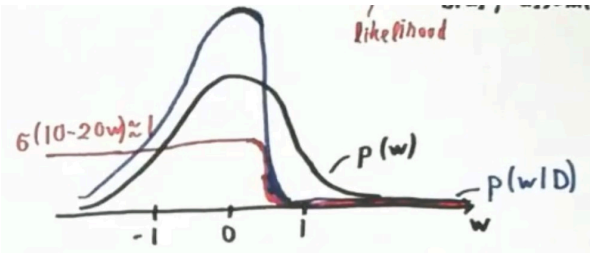
posterior p(w | D) is not Gaussian, but can approximate it with a Gaussian

e.g. known prior, known 1 datapoint (y = 1 at x =–20, bias=10)

since weight isn't a large positive value, slide off positive region & renormalize (area under curve sums to 1)

$$p(w) \propto \mathcal{N}(w; 0, 1)$$
$$p(w \mid \mathcal{D}) \propto \mathcal{N}(w; 0, 1)\, \sigma(10 - 20w)$$

Normal curve: prior; Red: sliced off, before normalize; Black: after normalized



Since posterior = sigmoid multiplied,

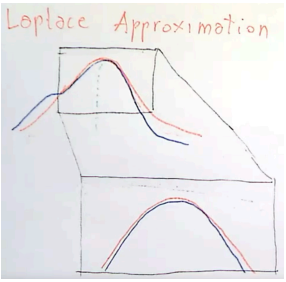
every time multiply the posterior by a sigmoidal likelihood, softly carve away half of the weight space in some direction

Bayesian central limit theorem: after many datapoints, distribution over plausible weights looks gaussian

Week 10b Laplace approximation

approximate a distribution P (w | D) (not exactly gaussian) with a Gaussian

incremental improvement of the MAP approximation to Bayesian inference



matches mode (of Gaussian approximation = mode of posterior w*)

matches the **curvature** of the log probability density at location i,j (Hessian)

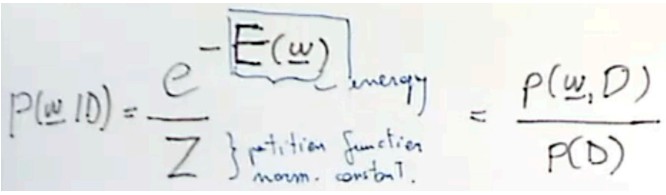
=> goal: find mode & second derivative

Approximate posterior by Boltzmann distribution

positive, so e; integrate to one, so divide by normalization constant (partition function) Z

Energy E

Z = P(D); e^(-E(w)) = p(w,D)



Energy E: complex function we try to approximate (-log P(w,D))

most probable weight **w***

= maximize posterior (MAP)

proportional to energy, drop other terms (Z and exp)

= minimum of energy (since exp is an increasing function):

(minimum = turning point, first derivative = 0, second derivative = curvature)

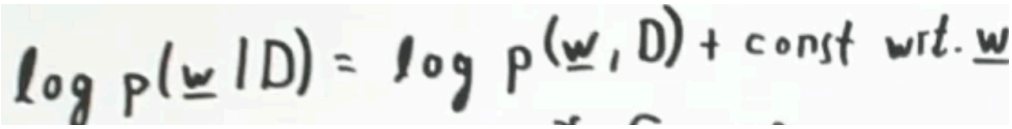
w* = L2 regularization or MAP fit

$$E(\mathbf{w}) = -\log p(\mathbf{w}, \mathcal{D}), \qquad \mathbf{w}^* = \arg\min_{\mathbf{w}} E(\mathbf{w}).$$

Take log of posterior (put in MAP)

By property of conditional probability, P(D) = some constant w.r.t w:

since log of distribution is closer to quadratic function (good for optimizer)

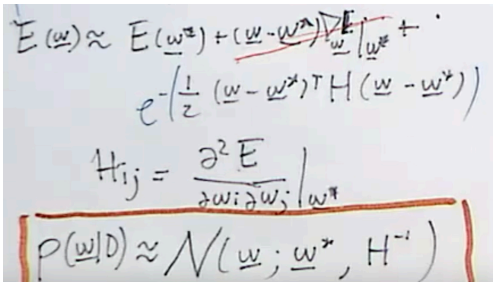


Energy being approximated by

its value observed in local points E(w*)

+ gradient, evaluated at local point

+ quadratic term (Taylor approximation)



Hessian, second derivative at the optimum, w = w* (by MAP):

= curvature of the function

= how sharply the distribution is peaked in different directions

= how spread out the gaussian would be

$$H_{ij} = \left. \frac{\partial^2 E(\mathbf{w})}{\partial w_i \partial w_j} \right|_{\mathbf{w}=\mathbf{w}^*}$$

E(w) approx. by Gaussian, with N (μ , Σ),
energy = -log P(w,D) is:

$$E_{\mathcal{N}}(\mathbf{w}) = \frac{1}{2}(\mathbf{w}-\boldsymbol{\mu})^{\top}\Sigma^{-1}(\mathbf{w}-\boldsymbol{\mu}),$$

So w*= μ and H = Σ^(–1), covariance is Σ = H^(–1)

Laplace approximation to the posterior distribution (by matching minimum & curvature):
When prior & likelihood are gaussian, L approx is exact

$$p(\mathbf{w} \mid \mathcal{D}) \approx \mathcal{N}(\mathbf{w};\, \mathbf{w}^*, H^{-1})$$

Approximate marginal likelihood P (D | M) (= partition function Z)
Expand the Gaussian PDF:

$$p(\mathbf{w} \mid \mathcal{D}) = \frac{p(\mathbf{w},\mathcal{D})}{P(\mathcal{D})} \approx \mathcal{N}(\mathbf{w};\, \mathbf{w}^*, H^{-1}) = \frac{|H|^{1/2}}{(2\pi)^{D/2}} \exp\left(-\frac{1}{2}(\mathbf{w}-\mathbf{w}^*)^{\top} H (\mathbf{w}-\mathbf{w}^*)\right)$$

In order to get normalizing constant P(D), does not depends on w, could plug in any values for w,
if plug in w* (w=w*), exp term disappears, very convenient

get P(D):
get w* by SGD on negative log likelihood with regularization term; = MAP
DIFFERENTIATE: D in P(D): training data; D in D/2: # of parameters w

$$\frac{p(\mathbf{w}^*,\mathcal{D})}{P(\mathcal{D})} \approx \frac{|H|^{1/2}}{(2\pi)^{D/2}}, \qquad \boxed{P(\mathcal{D}) \approx \frac{p(\mathbf{w}^*,\mathcal{D})(2\pi)^{D/2}}{|H|^{1/2}}}.$$

Approximate p(D) for different models, choose the model with the highest marginal likelihood P(D | M)
since it is approximation, can go wrong if approx. by gaussian is poor fit

Prediction
Given test input & observations, want to take our belief about different ws into account

but distribution in integral (eq bellow) is not gaussian anymore (different from Gaussian linear reg)

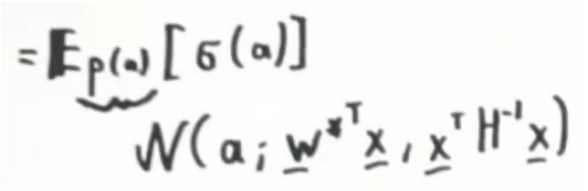
$$P(y \mid \mathbf{x}, \mathcal{D}) = \int P(y \mid \mathbf{x}, \mathbf{w})\, p(\mathbf{w} \mid \mathcal{D})\, \mathrm{d}\mathbf{w}$$

P (y | x, w) = sigmoid, since logistic regression;
use a normal distribution to weight the sigmoid

$$P(y=1 \mid \mathbf{x}, \mathcal{D}) \approx \int \sigma(\mathbf{w}^\top \mathbf{x})\, \mathcal{N}(\mathbf{w};\, \mathbf{w}^*, H^{-1})\, \mathrm{d}\mathbf{w}$$
$$= \mathbb{E}_{\mathcal{N}(\mathbf{w};\, \mathbf{w}^*, H^{-1})} \left[\sigma(\mathbf{w}^\top \mathbf{x}) \right].$$

Can not compute high dim integral,
Goal: convert to 1D, transform sigma to gaussian (to get closed form solution)

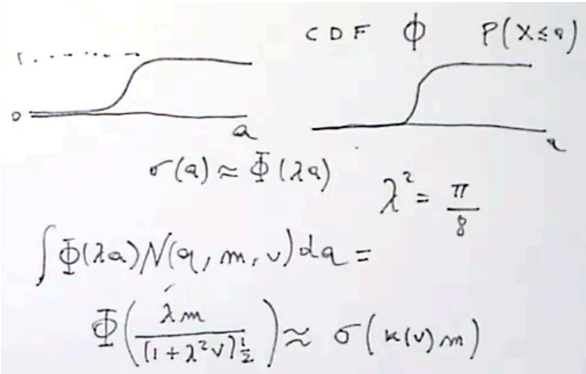
since w^T.X is scalar, convert to 1D integral
scalar a = w^T.X; p(a) = gaussian;
a is linear trans of the distribution of w by X
take the expectation over this scalar quantity



prediction = 1D integral, could compute numerically

$$P(y=1 \mid \mathbf{x}, \mathcal{D}) \approx \mathbb{E}_{\mathcal{N}(a;\, \mathbf{w}^{*\top}\mathbf{x},\, \mathbf{x}^\top H^{-1}\mathbf{x})} \left[\sigma(a) \right]$$
$$= \int \sigma(a)\, \mathcal{N}(a;\, \mathbf{w}^{*\top}\mathbf{x},\, \mathbf{x}^\top H^{-1}\mathbf{x})\, \mathrm{d}a.$$

Probit approximation: approx. sigmoid by gaussian CDF
Since we know how to compute and integrate out CDF of gaussian * gaussian
lambda: adjust the slopes
term inside sigmoid: kappa; m = w*^T.x



Kappa of covariance k(v):

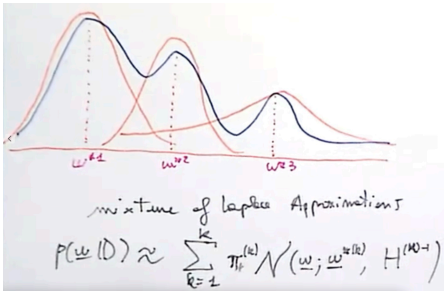
predictions use the most probable or MAP weights w*

BUT w*Tx term will be scaled down by kappa when uncertain about activation

=> predictions will be less confident far from the data

$$P(y=1 \mid \mathbf{x}, \mathcal{D}) \approx \sigma(\kappa\, \mathbf{w}^{*\top} \mathbf{x}), \qquad \kappa = \frac{1}{\sqrt{1 + \frac{\pi}{8} \mathbf{x}^\top H^{-1} \mathbf{x}}}.$$

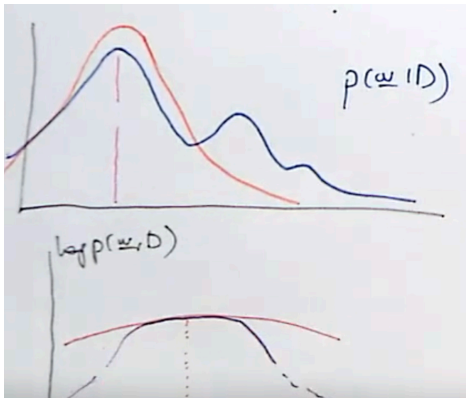
Mixture (sum) of Laplace approx. by coefficient pi



Is the Laplace approximation reasonable?

Laplace approximation failure

when distribution is not Gaussian, or multi-modal (posterior for neural net)


p(w|D)
log p(w,D)