Today

- Final exam does not include this nor next week’s lecture.
- Continuing in our theme of probabilistic models for continuous variables.
  - Probabilistic interpretation of linear regression
  - Probabilistic interpretation of PCA
- (Optional) Bayesian model selection.
Completing the Square for Gaussians

- First, we’re going to review a very powerful technique that will let us figure out the distribution of Gaussian random variables.
- It’s a multivariate generalization of completing the square.
- The density of \( \mathcal{N}(\mu, \Sigma) \) satisfies:

\[
\log p(x) = -\frac{1}{2} (x - \mu)^\top \Sigma^{-1} (x - \mu) + \text{const}
\]

\[
= -\frac{1}{2} x^\top \Sigma^{-1} x + x^\top \Sigma^{-1} \mu + \text{const}
\]

- Thus, if we know \( w \) is Gaussian with \textit{unknown} mean and covariance, and we also know that

\[
\log p(w) = -\frac{1}{2} w^\top A w + w^\top b + \text{const}
\]

for \( A \) positive definite, then we know that

\[
w \sim \mathcal{N}(A^{-1}b, A^{-1})
\]
We’re going to be Bayesian about the parameters of the model.
  ▶ This is in contrast with naïve Bayes and GDA: in those cases, we used Bayes’ rule to infer the class, but used point estimates of the parameters.
  ▶ By inferring a posterior distribution over the parameters, the model can know what it doesn’t know.

How can uncertainty in the predictions help us?
  ▶ Smooth out the predictions by averaging over lots of plausible explanations (just like ensembles!)
  ▶ Assign confidences to predictions
  ▶ Make more robust decisions
Recap: Linear Regression

- Given a training set of inputs and targets \( \{(x^{(i)}, t^{(i)})\}_{i=1}^N \)
- Linear model:
  \[
y = \mathbf{w}^\top \psi(x)
\]
- Vectorized, we have the design matrix \( \mathbf{X} \) in input space and

\[
\psi = \begin{bmatrix}
- \psi(x^{(1)}) & - \\
- \psi(x^{(2)}) & - \\
\vdots & \\
- \psi(x^{(N)}) & -
\end{bmatrix}
\]

and predictions

\[
y = \mathbf{\Psi} \mathbf{w}
\]
Recap: Linear Regression

- Squared error loss:
  \[ L(y, t) = \frac{1}{2} \|y - t\|^2 \]
- \( L_2 \) regularization:
  \[ \phi(w) = \frac{\lambda}{2} \|w\|^2 \]

- Solution 1: solve analytically by setting the gradient to 0
  \[ w = (\Psi^\top \Psi + \lambda I)^{-1} \Psi^\top t \]
- Solution 2: solve approximately using gradient descent
  \[ w \leftarrow (1 - \alpha \lambda)w - \alpha \Psi^\top (y - t) \]
We can give linear regression a probabilistic interpretation by assuming a Gaussian noise model:

\[ t \mid x \sim \mathcal{N}(w^\top \psi(x), \sigma^2) \]

Linear regression is just maximum likelihood under this model:

\[
\frac{1}{N} \sum_{i=1}^{N} \log p(t^{(i)} \mid x^{(i)}; w, b) = \frac{1}{N} \sum_{i=1}^{N} \log \mathcal{N}(t^{(i)}; w^\top \psi(x^{(i)}), \sigma^2)
\]

\[
= \frac{1}{N} \sum_{i=1}^{N} \log \left[ \frac{1}{\sqrt{2\pi\sigma}} \exp \left( -\frac{(t^{(i)} - w^\top \psi(x^{(i)}))^2}{2\sigma^2} \right) \right]
\]

\[
= \text{const} - \frac{1}{2N\sigma^2} \sum_{i=1}^{N} (t^{(i)} - w^\top \psi(x^{(i)}))^2
\]
We can view an $L_2$ regularizer as MAP inference with a Gaussian prior.

Recall MAP inference:

$$\arg \max_w \log p(w \mid D) = \arg \max_w \left[ \log p(w) + \log p(D \mid w) \right]$$

We just derived the likelihood term $\log p(D \mid w)$:

$$\log p(D \mid w) = \text{const} - \frac{1}{2N\sigma^2} \sum_{i=1}^{N} (t^{(i)} - w^\top \psi(x^{(i)}))^2$$

Assume a Gaussian prior, $w \sim \mathcal{N}(m, S)$:

$$\log p(w) = \log \mathcal{N}(w; m, S)$$

$$= \log \left[ \frac{1}{(2\pi)^{D/2} |S|^{1/2}} \exp \left( -\frac{1}{2} (w - m)^\top S^{-1} (w - m) \right) \right]$$

$$= -\frac{1}{2} (w - m)^\top S^{-1} (w - m) + \text{const}$$

Commonly, $m = 0$ and $S = \eta I$, so

$$\log p(w) = -\frac{1}{2\eta} \|w\|^2 + \text{const}.$$  

This is just $L_2$ regularization!
Recap: Full Bayesian Inference

- Recall: full Bayesian inference makes predictions by averaging over all likely explanations under the posterior distribution.
- Compute posterior using Bayes’ Rule:
  \[ p(w \mid D) \propto p(w)p(D \mid w) \]
- Make predictions using the posterior predictive distribution:
  \[ p(t \mid x, D) = \int p(w \mid D) p(t \mid x, w) \, dw \]
- Doing this lets us quantify our uncertainty.
Prior distribution: \( w \sim \mathcal{N}(0, S) \)

Likelihood: \( t \mid x, w \sim \mathcal{N}(w^\top \psi(x), \sigma^2) \)

Assuming fixed/known \( S \) and \( \sigma^2 \) is a big assumption. More on this later.
Bayesian linear regression considers various plausible explanations for how the data were generated.

It makes predictions using all possible regression weights, weighted by their posterior probability.

Here are samples from the prior $p(w)$ and posteriors $p(w | D)$.
Deriving the posterior distribution:

\[
\log p(w | D) = \log p(w) + \log p(D | w) + \text{const}
\]

\[
= -\frac{1}{2} w^\top S^{-1} w - \frac{1}{2\sigma^2} \|\Psi w - t\|^2 + \text{const}
\]

\[
= -\frac{1}{2} w^\top S^{-1} w - \frac{1}{2\sigma^2} \left( w^\top \Psi^\top \Psi w - 2t^\top \Psi w + t^\top t \right) + \text{const}
\]

\[
= -\frac{1}{2} w^\top \left( \sigma^{-2} \Psi^\top \Psi + S^{-1} \right) w - \frac{1}{\sigma^2} t^\top \Psi w + \text{const} \quad \text{(complete the square!)}
\]

Thus \( w | D \sim \mathcal{N}(\mu, \Sigma) \) where

\[
\mu = \sigma^{-2} \Psi^\top t
\]

\[
\Sigma = \left( \sigma^{-2} \Psi^\top \Psi + S^{-1} \right)^{-1}
\]
Since a Gaussian prior leads to a Gaussian posterior, this means the Gaussian distribution is the conjugate prior for linear regression!

Compare $\mu$ the closed-form solution for linear regression:

$$w = (\Psi^{\top} \Psi + \lambda I)^{-1} \Psi^{\top} t$$

This is the mean of the posterior, assuming that $S = \lambda^{-1} I$ and $\sigma = 1$.

$\lambda^{-1}$ is the standard deviation of the prior. As this becomes infinite, the mean of the posterior converges to the maximum likelihood solution.
Bayesian Linear Regression

— Bishop, Pattern Recognition and Machine Learning
Example with radial basis function (RBF) features

\[ \psi_j(x) = \exp \left( -\frac{(x - \mu_j)^2}{2s^2} \right) \]

— Bishop, Pattern Recognition and Machine Learning
Bayesian Linear Regression

Functions sampled from the posterior:
Bayesian Linear Regression

- The posterior just gives us distribution over the parameter space, but if we want to make predictions, the natural choice is to use the posterior predictive distribution.

- Posterior predictive distribution:

\[
p(t \mid x, \mathcal{D}) = \int \underbrace{p(t \mid x, w)}_{\mathcal{N}(t \mid w^\top \psi(x), \sigma)} \underbrace{p(w \mid \mathcal{D})}_{\mathcal{N}(w; \mu, \Sigma)} \, dw
\]

- Another interpretation: \( t = w^\top \psi(x) + \varepsilon \), where \( \varepsilon \sim \mathcal{N}(0, \sigma) \) is independent of \( w \mid \mathcal{D} \sim \mathcal{N}(\mu, \Sigma) \).
Another interpretation:  \( t = w^\top \psi(x) + \varepsilon \), where \( \varepsilon \sim \mathcal{N}(0, \sigma) \) is independent of \( w \mid D \sim \mathcal{N}(\mu, \Sigma) \).

By the linear combination rules for Gaussian random variables, \( t \) is a Gaussian distribution with parameters

\[
\mu_{\text{pred}} = \mu^\top \psi(x) \\
\sigma^2_{\text{pred}} = \psi(x)^\top \Sigma \psi(x) + \sigma^2
\]

Hence, the posterior predictive distribution is \( \mathcal{N}(t ; \mu_{\text{pred}}, \sigma^2_{\text{pred}}) \).
Bayesian Linear Regression

Here we visualize confidence intervals based on the posterior predictive mean and variance at each point:

---

_Bishop, Pattern Recognition and Machine Learning_
The formulation of PCA that we saw earlier in the course was motivated heuristically.

We will show that it can be expressed as the maximum likelihood estimate of a certain probabilistic model.
Recall: PCA

- Data set \( \{x^{(i)}\}_{i=1}^N \)
- Each input vector \( x^{(i)} \in \mathbb{R}^D \) is approximated as \( \hat{\mu} + Uz^{(i)} \),
  \[ x^{(i)} \approx \tilde{x}^{(i)} = \hat{\mu} + Uz^{(i)} \]
  where \( \hat{\mu} = \frac{1}{n} \sum_i x^{(i)} \) is the data mean, \( U \in \mathbb{R}^{D \times K} \) is the orthogonal basis for the principal subspace, and \( z^{(i)} \in \mathbb{R}^K \) is the code vector
  \[ z^{(i)} = U^T (x^{(i)} - \hat{\mu}) \]
- \( U \) is chosen to minimize the reconstruction error
  \[ U^* = \arg \min_U \sum_i \| x^{(i)} - \hat{\mu} + UU^T (x^{(i)} - \hat{\mu}) \|^2 \]
To formulate probabilistic PCA, let’s start with a latent variable model.

Similar to the Gaussian mixture model, but we will assume continuous, Gaussian latents:

\[ z \sim \mathcal{N}(0, I) \]

\[ x \mid z \sim \mathcal{N}(Wz + \mu, \sigma^2 I) \]

Note: this is a naive Bayes model, because \( p(x \mid z) \) factorizes with respect to the dimensions of \( x \).

What sort of data does this model produce?
- \( z \) is a random coordinate within the affine space centered at \( \mu \) and spanned by the columns of \( W \).
- To get the random variable \( x \), we sample a standard Normal \( z \) and then add a small amount of isotropic noise to \( Wz + \mu \).
Probabilistic PCA

$p(z)\quad \hat{z}$

$p(x|\hat{z})$

$\mu$

$\hat{z}|w|$

$p(x)$

— Bishop, Pattern Recognition and Machine Learning
To perform maximum likelihood in this model, we need to maximize the following:

$$\max_{W, \mu, \sigma^2} \log p(x | W, \mu, \sigma^2) = \max_{W, \mu, \sigma^2} \log \int p(x | z, W, \mu, \sigma^2)p(z) \, dz$$

This was hard for the Gaussian mixture model, but in this case it’s easy.

$p(x | W, \mu, \sigma^2)$ will be Gaussian (confirm this) and so we just need to compute and Cov[$x$] and $E[x]$. 


\[ \mathbb{E}[x] = \mathbb{E}[Wz + \mu + \epsilon] = \mu \]

\[ \text{Cov}[x] = \mathbb{E}[(Wz + \epsilon)(Wz + \epsilon)^\top] \]
\[ = \mathbb{E}[(Wzz^\top W^\top] + \text{Cov}[\epsilon\epsilon^\top] \]
\[ = WW^\top + \sigma^2 I \]
Thus, the likelihood of the data under this model is given by

$$- \frac{ND}{2} \log(2\pi) - \frac{N}{2} \log |\mathbf{C}| - \frac{1}{2} \sum_{i=1}^{N} (\mathbf{x}^{(i)} - \boldsymbol{\mu})^\top \mathbf{C}^{-1} (\mathbf{x}^{(i)} - \boldsymbol{\mu})$$

where $\mathbf{C} = \mathbf{W} \mathbf{W}^\top + \sigma^2 \mathbf{I}$.

It’s a bit involved to derive the maximum likelihood solution, so we will skip it, but Tipping and Bishop (Probabilistic PCA, 1999) show that this is maximized at the following stationary points.
\[ \hat{\mu}_{\text{MLE}} = \frac{1}{N} \sum_{i=1}^{N} x^{(i)} \]
\[ \hat{W}_{\text{MLE}} = \hat{U}_{\text{MLE}}(\hat{L}_{\text{MLE}} - \hat{\sigma}^2_{\text{MLE}}I)^{\frac{1}{2}}R \]

where \( \hat{U}_{\text{MLE}} \) is the matrix whose columns are the \( K \) unit eigenvectors of the empirical covariance matrix \( \hat{\Sigma} \) that have the largest eigenvalues, \( \hat{L}_{\text{MLE}} \in \mathbb{R}^{K \times K} \) is the diagonal matrix whose elements are the corresponding eigenvalues, and \( R \) is any orthogonal matrix.
Probabilistic PCA: Maximum Likelihood

\[ \hat{\sigma}_{\text{MLE}}^2 = \frac{1}{D - K} \sum_{i=K+1}^{D} \lambda_i \]

where \( \lambda_i \) is the \( i \)th largest eigenvalue of the empirical covariance matrix \( \hat{\Sigma} \) of the data. In otherwords, the average variance of the discarded subspace.
That seems complex, to get an intuition about how this model behaves when it is fit to data, let's consider the MLE density.

Recall that the marginal distribution on $x$ in our fitted model is a Gaussian with mean

$$\hat{\mu}_{\text{MLE}}$$

and covariance

$$\hat{W}_{\text{MLE}} \hat{W}_{\text{MLE}}^T + \hat{\sigma}_{\text{MLE}}^2 I = \hat{U}_{\text{MLE}} (\hat{L}_{\text{MLE}} - \hat{\sigma}_{\text{MLE}}^2 I) \hat{U}_{\text{MLE}}^T + \hat{\sigma}_{\text{MLE}}^2 I$$

The covariance gives us a nice intuition about the type of model this forms.
Consider centering the data and checking the variance along one of the unit eigenvectors \( u_i \), which are the eigenvectors forming the columns of \( \hat{U}_{\text{MLE}} \):

\[
\text{Var}(u_i^T (x - \hat{\mu}_{\text{MLE}})) = u_i^T \text{Cov}[x] u_i \\
= u_i^T \hat{U}_{\text{MLE}} (\hat{L}_{\text{MLE}} - \hat{\sigma}_{\text{MLE}}^2 I) \hat{U}_{\text{MLE}}^T u_i + \hat{\sigma}_{\text{MLE}}^2 \\
= \lambda_i - \hat{\sigma}_{\text{MLE}}^2 + \hat{\sigma}_{\text{MLE}}^2 = \lambda_i
\]

Now, consider centering the data and checking the variance along any unit vector orthogonal to the subspace spanned by \( \hat{U}_{\text{MLE}} \):

\[
\text{Var}(u_i^T (x - \hat{\mu}_{\text{MLE}})) = u_i^T \hat{U}_{\text{MLE}} (\hat{L}_{\text{MLE}} - \hat{\sigma}_{\text{MLE}}^2 I) \hat{U}_{\text{MLE}}^T u_i + \hat{\sigma}_{\text{MLE}}^2 \\
= \hat{\sigma}_{\text{MLE}}^2
\]

In other words, the model captures the variance along the principle axes and approximates the variance in all remaining directions with a single variance.
Probably easier to visualize after implementing it.
How does it relate to PCA?

- The posterior mean is given by:
  \[
  \mathbb{E}[z | x] = \left( \hat{W}_{\text{MLE}}^T \hat{W}_{\text{MLE}} + \hat{\sigma}^2_{\text{MLE}} I \right)^{-1} \hat{W}_{\text{MLE}}^T(x - \hat{\mu}_{\text{MLE}})
  \]

- So, if we don’t fit \( \sigma^2 \) and instead take it to 0 we get:
  \[
  \mathbb{E}[z | x] \xrightarrow{\sigma^2 \to 0} \left( \hat{W}_{\text{MLE}}^T \hat{W}_{\text{MLE}} \right)^{-1} \hat{W}_{\text{MLE}}^T(x - \hat{\mu}_{\text{MLE}})
  \]

- Can show that this is a projection onto an affine space spanned by the columns of \( \hat{U}_{\text{MLE}} \).
Why Probabilistic PCA (PPCA)?

- Fitting a full-covariance Gaussian model of data requires $D(D + 1)/2 + D$ parameters. With PPCA we model only the $K$ most significant correlations and this only requires $O(D)$ parameters as long as $K$ is small.
- Basis of Bayesian treatment of PCA, which gives us a Bayesian method for determining the dimensionality of the principal subspace (i.e. $K$).
- Existence of likelihood functions allows direct comparison with other probabilistic models.
- Can use PPCA as a class-conditional density (as in GDA) to reduce the requirement to fit and store $O(D^2)$ parameters.
Recap: Gaussian models that we covered

- Gaussian discriminant analysis.
  - Gaussian class-conditional generative model $p(x \mid t)$ used for classification.

- Gaussian mixture model.
  - Gaussian latent variable model $p(x) = \sum_z p(x, z)$ used for clustering.

- Bayesian linear regression.
  - Gaussian discriminative model $p(t \mid x)$ used for regression with a Bayesian analysis for the weights.

- Probabilistic PCA.
  - Gaussian latent variable model $p(x) = \int_z p(x, z)$ used for dimensionality reduction.
Optional material: Bayesian model selection
Occam’s Razor (optional)

- Consider selecting models from a Bayesian perspective.
- Related to Occam’s Razor: “Entities should not be multiplied beyond necessity.”
  - Named after the 14th century British theologian William of Occam
- Huge number of attempts to formalize mathematically
  - See Domingos, 1999, “The role of Occam’s Razor in knowledge discovery” for a skeptical overview.
- Common misinterpretation: your prior should favor simple explanations
- Better interpretation: by averaging over many hypothesis, Bayesian model selection naturally prefers simpler models.
Occam’s Razor (optional)

- Suppose you have a finite set of models, or hypotheses \( \{ \mathcal{H}_i \}_{i=1}^M \) (e.g. polynomials of different degrees)
- Posterior inference over models (Bayes’ Rule):

\[
p(\mathcal{H}_i \mid \mathcal{D}) \propto p(\mathcal{H}_i) p(\mathcal{D} \mid \mathcal{H}_i)
\]

The evidence is also called **marginal likelihood** since it requires marginalizing out the parameters:

\[
p(\mathcal{D} \mid \mathcal{H}_i) = \int p(\mathbf{w} \mid \mathcal{H}_i) p(\mathcal{D} \mid \mathbf{w}, \mathcal{H}_i) \, d\mathbf{w}
\]
Occam’s Razor (optional)

- $p(H_i)$ is typically uniform, so we can compare them based on marginal likelihood.

- **Bayesian model selection:**

\[
H^* = \operatorname{arg\,max}_i p(D | H_i)
\]

- What types of models does this procedure prefer?
Suppose $M_1$, $M_2$, and $M_3$ denote a linear, quadratic, and cubic model. $M_3$ is capable of explaining more datasets than $M_1$. But its distribution over $\mathcal{D}$ must integrate to 1, so it must assign lower probability to ones it can explain.
Occam’s Razor (optional)

- How does the evidence penalize complex models?

- Approximating the integral for $w \in \mathbb{R}$:

$$p(D | H_i) = \int p(D | w, H_i) p(w | H_i)$$

$$\approx \frac{p(D | w_{\text{MAP}}, H_i)}{\Delta w_{\text{prior}}} \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}}$$

best-fit likelihood

Occam factor
Occam’s Razor (optional)

Let’s investigate

\[ \log p(D \mid H_i) = \log p(D \mid w_{\text{MAP}}, H_i) + \log \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}} \]

- First term represents fit to the data given the most probable parameter values.
- Second term is a penalty, because it is negative \((\Delta w_{\text{posterior}} < \Delta w_{\text{prior}})\).
- Thus if the posterior is very peaked and confident about the data, this penalty term will be very negative.
**Occam’s Razor (optional)**

- \( w \in \mathbb{R}^M \) we have

\[
\log p(D \mid \mathcal{H}_i) = \log p(D \mid w_{\text{MAP}}, \mathcal{H}_i) + M \log \frac{\Delta w_{\text{posterior}}}{\Delta w_{\text{prior}}}
\]

- So the more parameters we have, the higher the penalty.
- Optimal model complexity is determined by a tradeoff.
- In Bayesian model selection, we naturally prefer simpler models that model the data well.