Our approach today:

- Incrementally update $J^T W J$, use dense methods.
  
  Pros: makes it "easy" to forget old things.
  Cons: can't relinearize. Sparsity will be compromised if we relinearize.

$$
\chi^2 = \sum_i (z_i(x) - \hat{z}_i) \Sigma_i^{-1} (z_i(x) - \hat{z}_i)
$$

Sum of squared errors.

$$
z_i(x) \approx \bar{J} \Delta x + b_i
$$

1st order Taylor expn.

so:

$$
z_i(x) - \hat{z}_i = \bar{J} \Delta x + b_i - \hat{z}_i
$$

let $r_i = z_i - \hat{z}_i$ (obs - predicted)

Note: $J_i$ is our hyperplane! A robot is a hyperplane collecting machine

$$
\chi^2 = \sum_i (\bar{J} \Delta x - r_i)^T \Sigma_i^{-1} (\bar{J} \Delta x - r_i)
$$

...(minimize by diff. w.r.t $\Delta x$, set to zero)...

$$
\left( \sum_i J_i^T \Sigma_i^{-1} J_i \right) \Delta x = \left( \sum_i J_i^T \Sigma_i^{-1} r_i \right)
$$

A matrix we can build up one hyperplane (or hyperplanes corresponding to 1 obs) at a time

A vector...

We will maintain 3 quantities:

- the info matrix
- the RHS
- $x$,
Structure of SLAM

Jacobian has a lot of structure.

Suppose we're in 2D $[\mathbf{x} \ \theta]$ -- what are the dimensions of the blocks & what do they mean?

$(3 \times 3, \ \Theta \text{ Jacobian of obs WRT variables})$

We can draw the SLAM graph from the Jacobian!

Information matrix -- sum of Int matrix of each eqn. Each eqn gives 4 non-zero blocks

$$J = \begin{bmatrix} A & B \\ A^T & B^T \end{bmatrix} \quad J^TJ = \begin{bmatrix} F \end{bmatrix}$$

$$J^TJ = \begin{bmatrix} A^T A & AB \\ BA & B^T B \end{bmatrix}$$

$$A = \frac{2z}{2x_i}, \quad B = \frac{2z}{2x_j}$$
Step 0.

Initialization. Approach 1: Include start state.

\[ x \sim N(0, 0) \]

↑ This is problematic in general. Suppose we're just "awfully sure". So:

\[ x \sim N(0, 0.01) \]

↑ Don't get carried away - condition # of matrix can get bad!

Approach 2: Skip initialization — anytime you have a reference to \( x_0 \), just plug in \( x_0 = 0 \).

(equiv. to computing \( p(x | x_0 = 0) \)

↑ All the state except \( x_0 \).

Pros: avoids singularity problems.

Cons: special cases all over (whenever ref state \( x_0 \)).

We'll use approach 1.
Step 1.

Odomory,

- Model: \( z = x_t - x_0 + w_1 = 1 \) Actual observed value.
  \( w_1 \sim N(0, \Sigma_1) = N(0, 1) \)

This is the first time \( x_t \) appears, Must Grow the state estimate.

\[
\begin{bmatrix}
4 \\
4
\end{bmatrix}
\]
old state (just 0 for us)

\[
\begin{bmatrix}
0 \\
1
\end{bmatrix}
\]
initial guess for \( x_t \)
(how about 1?)

\[
\text{enlarge info. matrix}
\]

\[
\begin{bmatrix}
(100) & 0 \\
0 & 0
\end{bmatrix}
\]

Exactly as though \( x_t \) was longer & \( J_t \)'s had extra zeros tacked on, At this point, it's rank-deficient.

- Now, add new hyperplanes by linearizing model WRT \( \Delta x \).

\[
\begin{bmatrix}
-1 & 1
\end{bmatrix}
\Delta x =
\begin{bmatrix}
-1 & 1
\end{bmatrix}
\text{obs initial guess (we got lucky!)}
\]

\[
J_t^T \Sigma^{-1} J_t = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}
\]
(assume \( \Sigma^{-1} = 1 \)
for laziness)

\[
\Sigma^{-1} = \begin{bmatrix}
101 & -1 \\
-1 & 1
\end{bmatrix}
\]

Similarly, update RHS: \( J_t^T \Sigma^{-1} r_t \)
Step 1, with cov. projection

Also could have done this via cov. projection!

\[ x' = f(x) = \begin{bmatrix} x_0 \\ 1 + x_0 - w_1 \end{bmatrix} \quad \text{solve obs. model for } x, \]

\[ E[x'] = E\left[ \begin{bmatrix} x_0 \\ 1 + x_0 - w_1 \end{bmatrix} \right] = E\left[ \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right] \]

\[ \text{VAR}[x'] = J_x^T \Sigma_x J_x^T + J_w^T \Sigma_w J_w^T \]

\[ J_x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \Sigma_x = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}, \quad J_w = \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad \Sigma_w = 1 \]

\[ = \begin{bmatrix} 100^{-1} & 100^{-1} \\ 100^{-1} & 100^{-1} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.01 & 0.01 \\ 0.01 & 1.01 \end{bmatrix} \]

\[ \Sigma_x' \]

And yes, \( \Sigma_x'^{-1} = \begin{bmatrix} 101 & -1 \\ -1 & 1 \end{bmatrix} \), from the other method,
Step 2

We observe a landmark.

Model: \( \mathbf{z}_k = f - x + w_z = 0.6 \quad w_z \sim N(0, 1) \)

\[ \mathbf{w} \uparrow \] our current position
\[ \downarrow \] landmark (feature) position

Just like step 1, this is our first reference to state element \( f \). Expand the state info:

\( \mathbf{x}' = \begin{bmatrix} 0 \\ 1 \\ \vdots \end{bmatrix} \) our previous state mean
\( \mathbf{x} \) initial guess for \( f \), (1.6.1)

\( \Sigma_{x'}^{-1} = \begin{bmatrix} \Sigma_{x}^{-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \) old \( \Sigma_x^{-1} \)

Add new hyperplane by linearizing WRT \( \mathbf{x}\):

\( \mathbf{J}_2 = \begin{bmatrix} 0 & -1 & 1 \end{bmatrix} \)\( \mathbf{r}_2 = \begin{bmatrix} 0 \end{bmatrix} \)

\[ \mathbf{J}_2 \Sigma_2^{-1} \mathbf{J}_2^T = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix} \]

So \( \Sigma^{-1} = \begin{bmatrix} 101 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix} \)

(consider \( \Sigma' \):

\[ \begin{bmatrix} 0.1 & 0.1 & 0.1 \\ 0.1 & 0.1 & 1.0 \\ 1.0 & 0.1 & 2.0 \end{bmatrix} \]

\( \Sigma' \) "inherited" the uncertainty of \( x_t \). Makes sense!
Step 3) Another adon, handled just like step 1.

\[ x = \begin{bmatrix} 0 \\ 1.6 \\ 2 \end{bmatrix} \quad \Sigma_x = \begin{bmatrix} 0.01 & 0.01 & 0.01 \\ 0.01 & 1.01 & 1.01 \\ 0.01 & 1.01 & 1.01 \end{bmatrix} \]

Step 4) We observe a feature.

\[ \text{Model} = f - x_2 = -0.5 \]
\[ \text{observed value} \]

\[ \text{Predicted location} = x_2 - 0.5 = 1.5. \]

Critical Question: Is this the same feature we saw before? A new feature?
Data Association

We've mostly assumed that we know which feature is which. What if we don't?

\[ \text{estimates of feature position} \]
\[ a \rightarrow f_1 \]
\[ \text{a} \& \text{b are obs without known feature association} \]
\[ f_2 \rightarrow \phi \]

Euclidean nearest-neighbor
- match each obs to closest landmark,
  + easy / fast
  - doesn't use uncertainties
- could a & b both be assoc. w the same feature?
  ▶ sometimes, this is sensible. Depends on sensor properties.
  ▶ when it's not, use greedy matching or stable marriage.

Is it a new landmark?

\[ \text{distance/error to best landmark.} \]
\[ \text{associate unsure, do nothing, create new landmark} \]

Greedy Matching
1. Consider all possible matches between A & B, \((n^2 \text{ distances})\).
2. Sort.
3. While (more matches)
   - \(m = \text{next best match,}\)
   - if \(m_a \text{ and } m_b \text{ have not yet been matched to any other features, accept } m\).
Doing Assoc "Right"

\[ p \left( \frac{z_1 \& z_2 \text{ are same object}}{p(\text{assoc})} \right) \]

\[ \int \int p(\text{assoc}, z, m) \; dz \; dm \]

\[ \int \int p(\text{assoc} \mid z, m) \; p(z|m) \; p(m) \; dz \; dm \]

Hopeless!

Forgetting

Marginalize