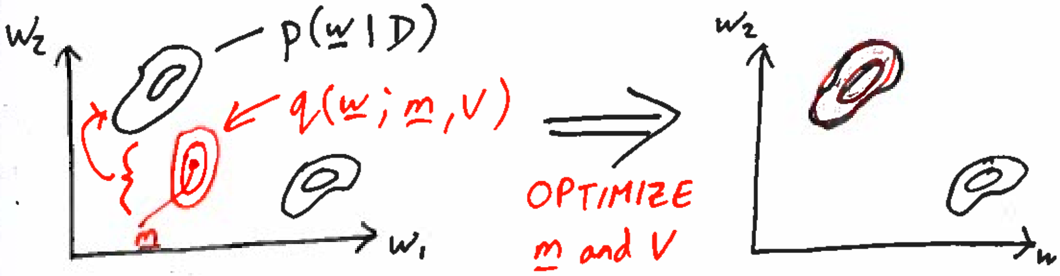
Variational Inference



Minimize $D_{KL}(q \,\|\, p(\underline{w}|D))$

$$p(\underline{w}|D) = \frac{p(D|\underline{w})\,p(\underline{w})}{p(D)}$$

$$D_{KL} = \mathbb{E}_q\left[\log \frac{q(\underline{w};\underline{m},V)}{p(\underline{w}|D)}\right] \geqslant 0$$

$$= \underbrace{\underbrace{-\mathbb{E}_q[\log p(D|\underline{w})]}_{\mathbb{E}[\text{neg. log likelihood}]} \underbrace{- \mathbb{E}_q[\log p(\underline{w})] + \overbrace{\mathbb{E}_q[\log q(\underline{w};\underline{m},V)]}^{-\text{Entropy}[q]}}_{D_{KL}(q\,\|\,p(\underline{w}))}}_{J} + \underbrace{\log p(D)}_{\text{Marginal Likelihood}}$$

$$= J + \log p(D) \geqslant 0 \quad \text{(Gibbs' inequality)}$$

Marginal Likelihood bound

$$\log p(D) \geqslant -J$$

Minimize J wrt $\underbrace{(\underline{m}, V)}_{\text{Approx. Posterior well.}}$ and $\underbrace{\sigma_w^2, \ldots}_{\text{Find good model}}$

Week 11 events

No MLPR lectures
(last lecture Thurs 21 Nov.)

Ed-Intelligence have two events!
(Links also in "week 11" of mlpr notes)

Mini NeurIPS

6pm Wed 27 Nov, AT LT2

tinyurl.com/mini-neurips-2019

Biases, failure + fairness in AI

6pm Fri 29 Nov, AT LT5

to-err-is-machine.eventbrite.co.u

Minimize J

Need tricks to make SGD work

Trick # 1, Reparameterize to make unconstrained

SGD on σ_w break, $\sigma_w < 0$ undefined

$\sigma_w \approx 0$ unstable

Set $\sigma_w = e^a$, optimize a

V positive definite, symmetric

Compute graph

$$\begin{tikzcd} & J & \\ \underline{m} \arrow[ur] & V \arrow[u] & \sigma_w = e^a \arrow[ul] \\ & L \arrow[u] & a \arrow[u] \\ & \tilde{L} \arrow[u] & \end{tikzcd}$$

$$V = LL^T$$

$$L_{ij} = \begin{cases} e^{\tilde{L}_{ii}} & i = j \\ \tilde{L}_{ij} & i > j \\ 0 & i < j \end{cases}$$

If could evaluate J,
backprop, do SGD on
$\underline{m}, \tilde{L}, a$

Evaluating the cost, J

$D_{KL}(q \,\|\, p(\underline{w}))$ this is "easy"

often Gaussian (Look it up in Matrix Cookbook)

Likelihood term

$$\mathbb{E}_q[\log p(D|\underline{w})]$$

$$= \mathbb{E}_q\Big[\sum_{n=1}^{N} \log p(y^{(n)}|\underline{x}^{(n)}, \underline{w})\Big]$$

For logistic regression $\rightarrow$ 1D integral

do numerically

Trick #2 "Reparameterization trick"

Stochastic estimate

$$\mathbb{E}_{N(\underline{w};\underline{m},V)}[f(\underline{w})] \approx \frac{1}{S}\sum_s f(\underline{w}^{(s)}) \quad \underline{w}^{(s)} \sim N(\cdots)$$

$$= \mathbb{E}_{N(\underline{\nu};\underline{0},\mathbb{I})}[f(L\underline{\nu}+\underline{m})]$$

$$\approx f(L\underline{\nu}+\underline{m}), \quad \underline{\nu} \sim N(\underline{0},\mathbb{I})$$

(Using only sample!)

Estimate of gradients

$$\nabla_{\underline{m}} \mathbb{E}_{N(\underline{w};\underline{m},V)}[f(\underline{w})]$$

$$\approx \nabla_{\underline{m}} f(L\underline{\nu}+\underline{m}) = \nabla_{\underline{w}} f(\underline{w})\Big|_{\underline{w}=L\underline{\nu}+\underline{m}}$$

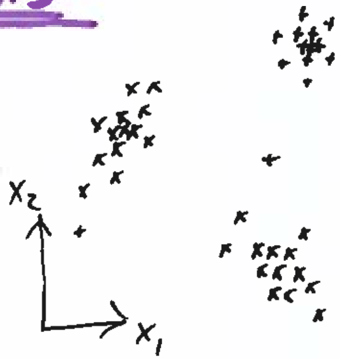
$$\nabla_{L} \mathbb{E}_{N(\underline{w};\underline{m},V)}[f(\underline{w})]$$

$$\approx \nabla_{\underline{L}} f(L\underline{\nu}+\underline{m})$$

$$= \nabla_{\underline{w}} f(\underline{w})\Big|_{\underline{w}=L\underline{\nu}+\underline{m}} \underline{\nu}^{T}$$

Use same $\nabla_{\underline{w}} f(\underline{w})$ derivatives as normal but on noisy weights.

Mixtures of Gaussians



Uses: - Clustering

- Model non-Gaussian distributions

Model:

Same as Gaussian Bayes classifier

Except "labels" are hidden or "latent" variables

Cluster choice $z^{(n)} \sim \text{Discrete}(\underline{\pi})$

$z^{(n)} \in \{1, 2, 3 \ldots K\}$

+ve vector sums to 1

Features

$\underline{x}^{(n)} \mid z^{(n)} = k \sim \mathcal{N}(\underline{x}^{(n)}; \underline{\mu}^{(k)}, \Sigma^{(k)})$

Likelihood

Given one observation: $p(\underline{x}^{(n)} \mid \theta = (\underline{\pi}, \{\underline{\mu}^{(k)}, \Sigma^{(k)}\}))$

$$= \sum_{k=1}^{K} p(\underline{x}^{(n)}, z^{(n)} = k \mid \theta)$$

$$= \sum_k p(\underline{x}^{(n)} \mid z^{(n)} = k, \theta)\, p(z^{(n)} = k \mid \theta)$$

$$= \sum_k \mathcal{N}(\underline{x}^{(n)}; \underline{\mu}^{(k)}, \Sigma^{(k)})\, \pi_k$$

Unsupervised learning, Clustering

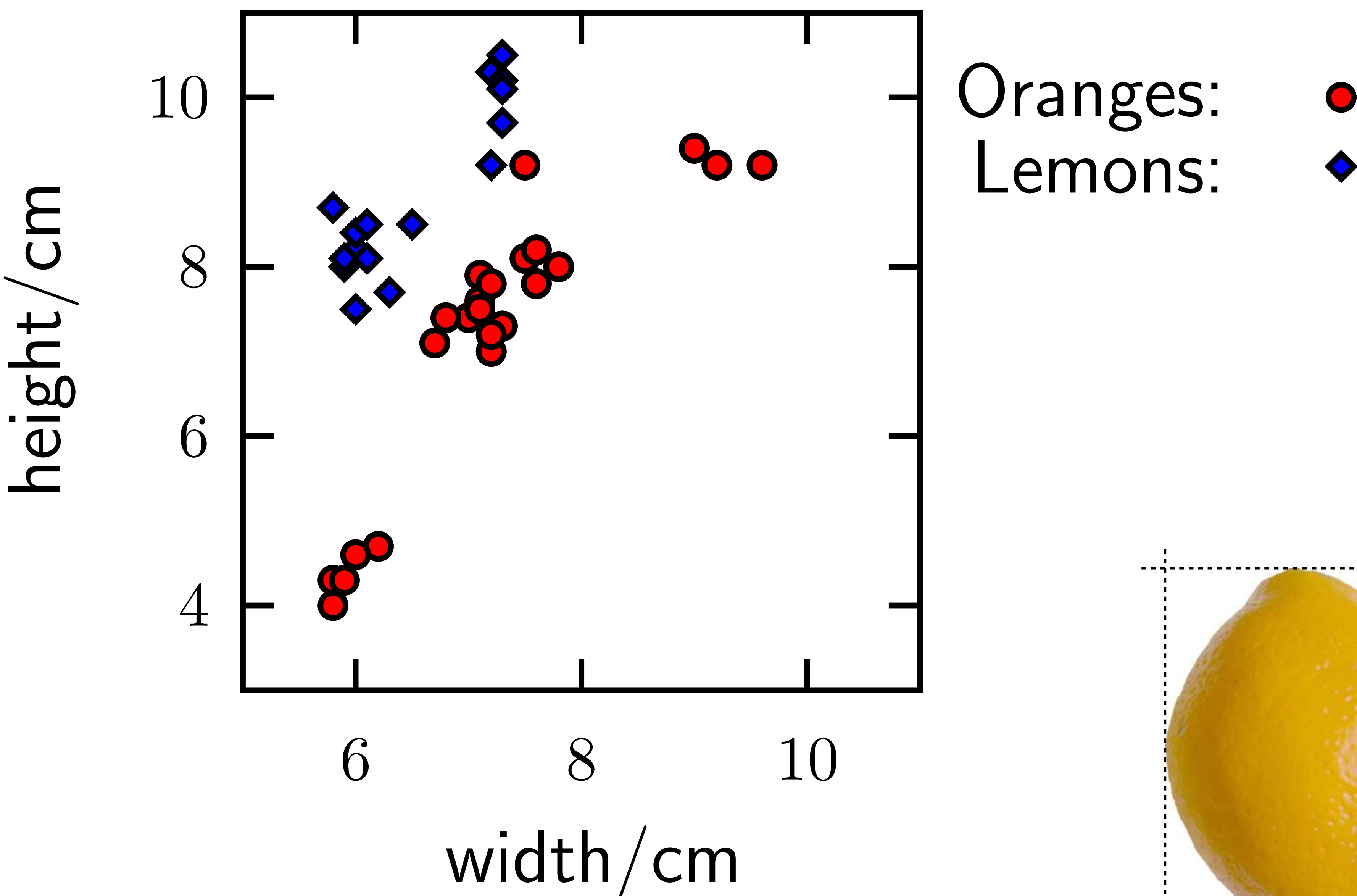
"Human brains are good at finding regularities in data.

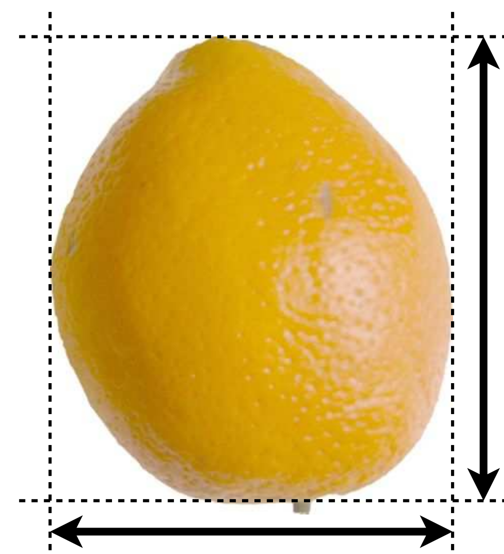
One way of expressing regularity is to put a set of objects into groups that are similar to each other.

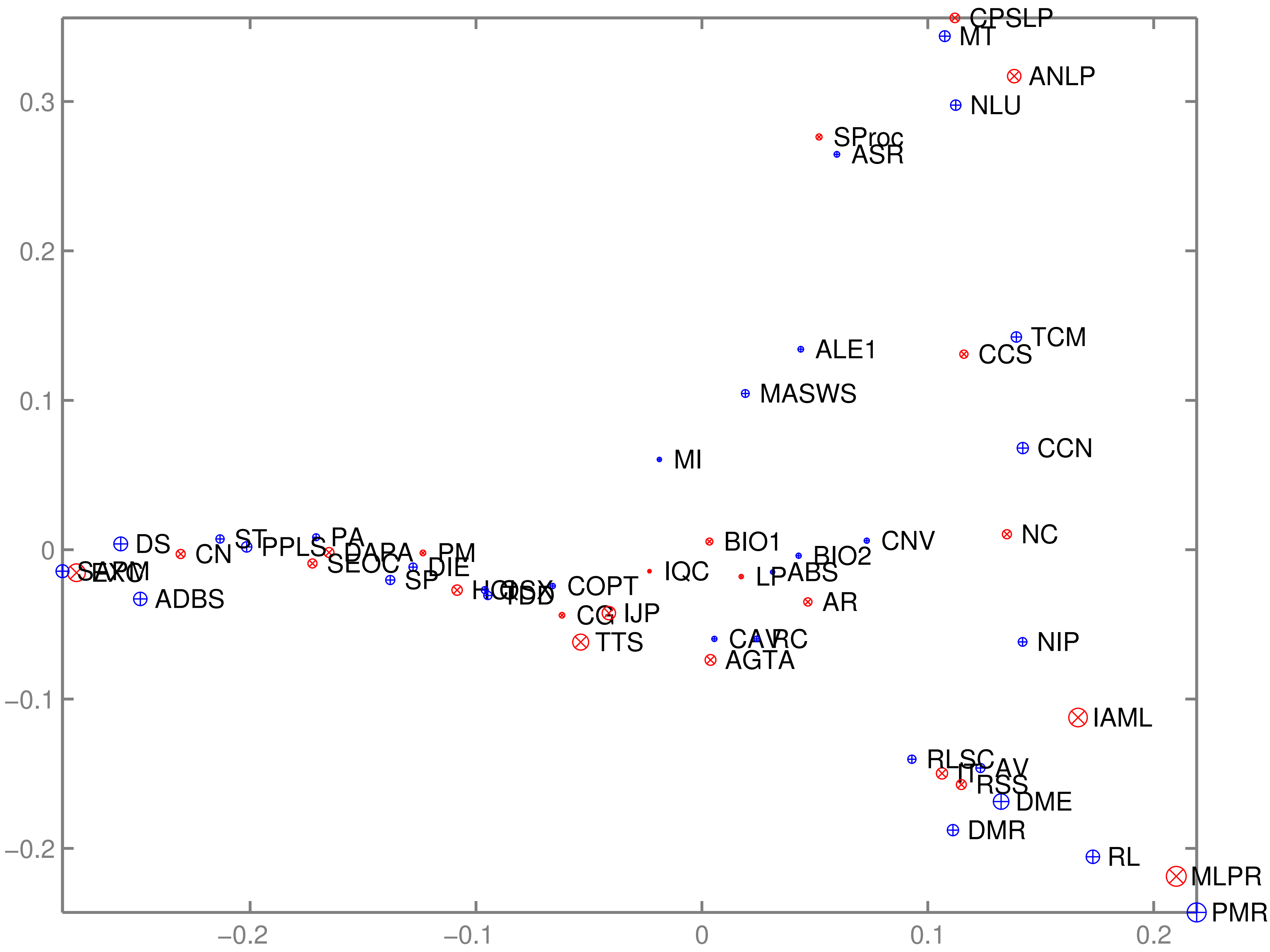
For example, biologists have found that most objects in the natural world fall into one of two categories: things that are brown and run away, and things that are green and don't run away. The first group they call animals, and the second, plants."

— David MacKay, ITILA textbook p284

Oranges and Lemons data







0.3
0.2
0.1
0
−0.1
−0.2
−0.2
−0.1
0
0.1
0.2
CPSLP
MT
ANLP
NLU
SProc
ASR
TCM
CCS
ALE1
MASWS
CCN
MI
DS
CN
ST
PPLS
PA
DAPA
PM
SEOC
DIE
SP
ADBS
COPT
TDD
CG
IJP
TTS
IQC
BIO1
BIO2
CNV
NC
LP
ABS
AR
RC
AGTA
NIP
IAML
RLSC
IT
AV
RSS
DME
DMR
RL
MLPR
PMR

Stanley

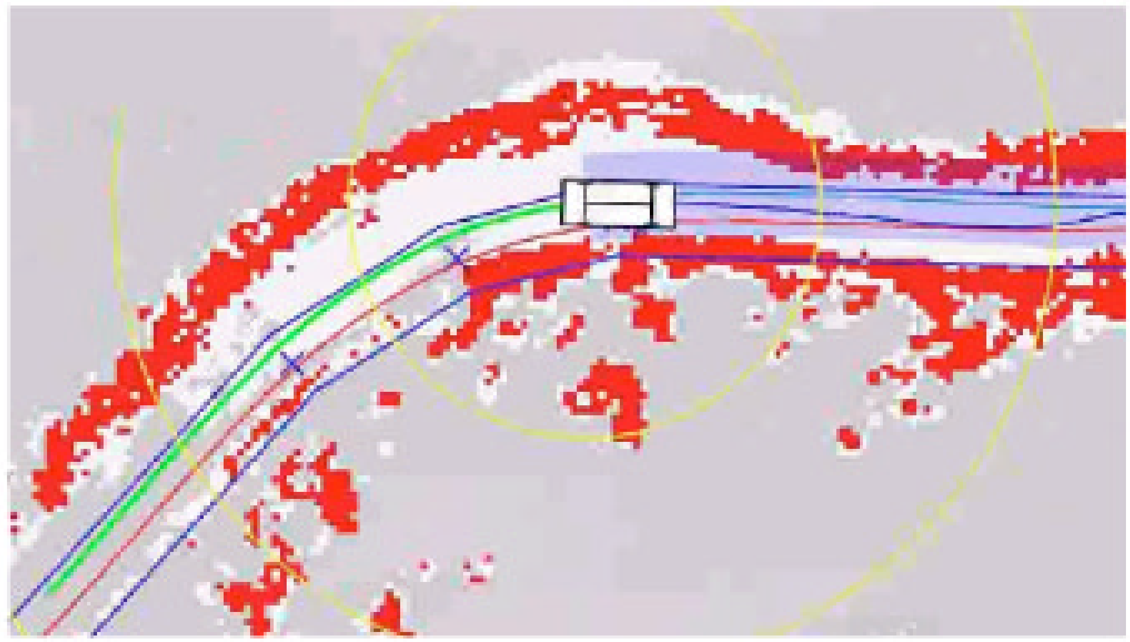
Stanford Racing Team; DARPA 2005 challenge

`http://robots.stanford.edu/talks/stanley/`

How to stay on a road?

Perception and intelligence

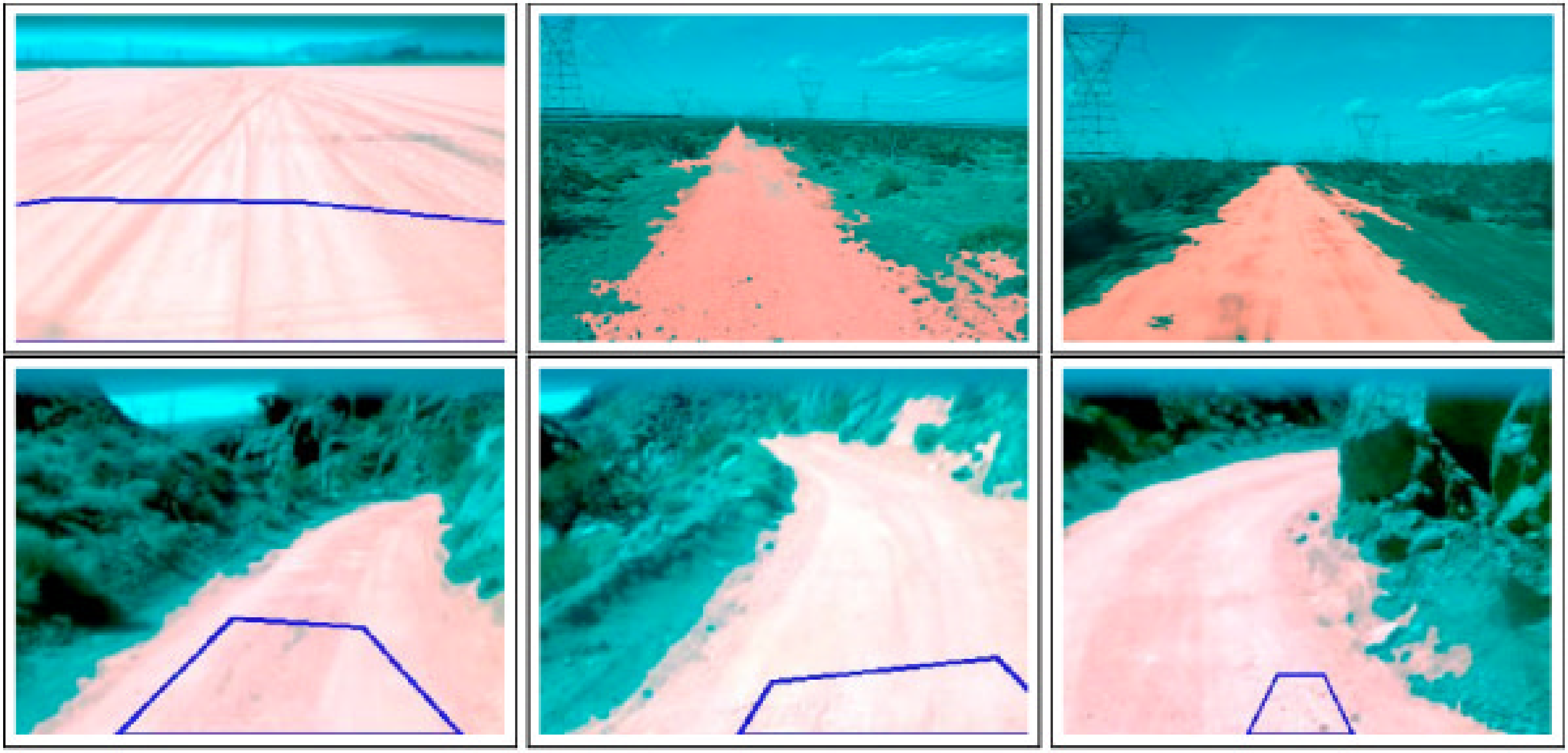
(a) **Beer Bottle Pass**



(b) **Map and GPS corridor**

It would look pretty stupid to run off the road,
just because the trip planner said so.

Clustering to stay on the road



Stanley used a Gaussian mixture model.
The cluster just in front is road (unless we already failed).