Note to other teachers and users of these slides. Andrew would be delighted if you found this source material useful in giving your own lectures. Feel free to use these slides verbatim, or to modify them to fit your own needs. PowerPoint originals are available. If you make use of a significant portion of these slides in your own lecture, please include this message, or the following link to the source repository of Andrew's tutorials: http://www.cs.cmu.edu/~awm/tutorials . Comments and corrections gratefully received.

Learning Gaussian Bayes Classifiers

Andrew W. Moore

Associate Professor

School of Computer Science

Carnegie Mellon University

www.cs.cmu.edu/~awm

awm@cs.cmu.edu

412-268-7599

Copyright © 2001, Andrew W. Moore

Sep 10th, 2001

Maximum Likelihood learning of Gaussians for Classification

- Why we should care
- 3 seconds to teach you a new learning algorithm
- What if there are 10,000 dimensions?
- What if there are categorical inputs?
- Examples “out the wazoo”

Copyright © 2001, Andrew W. Moore

Why we should care

- One of the original "Data Mining" algorithms
- Very simple and effective
- Demonstrates the usefulness of our earlier groundwork

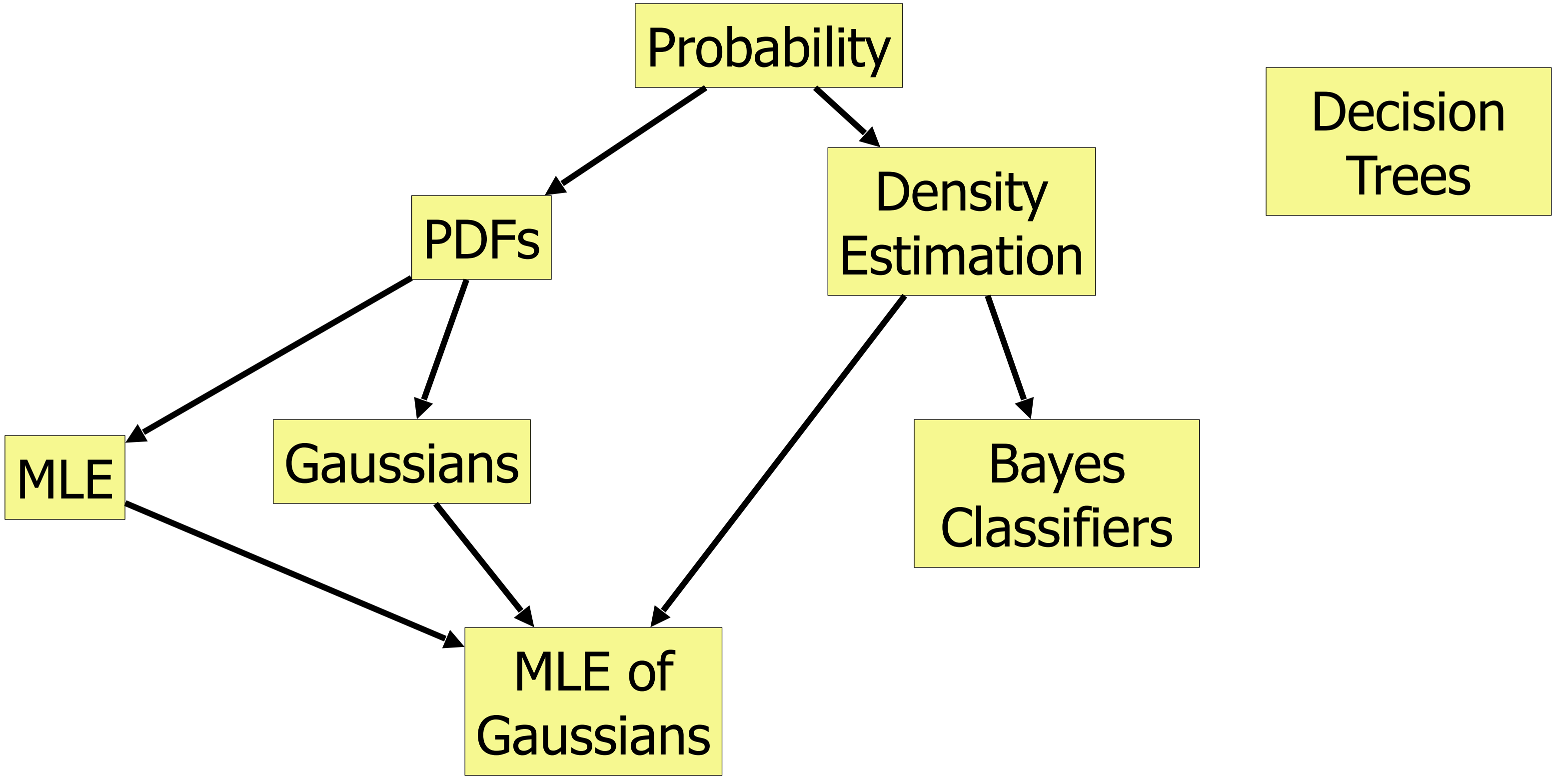
Where we were at the end of the MLE lecture…

	Categorical inputs only	Real-valued inputs only	Mixed Real / Cat okay
Inputs → Classifier → Predict category	Joint BC Naïve BC		Dec Tree
Inputs → Density Estimator → Prob-ability	Joint DE Naïve DE	Gauss DE	
Inputs → Regressor → Predict real no.			

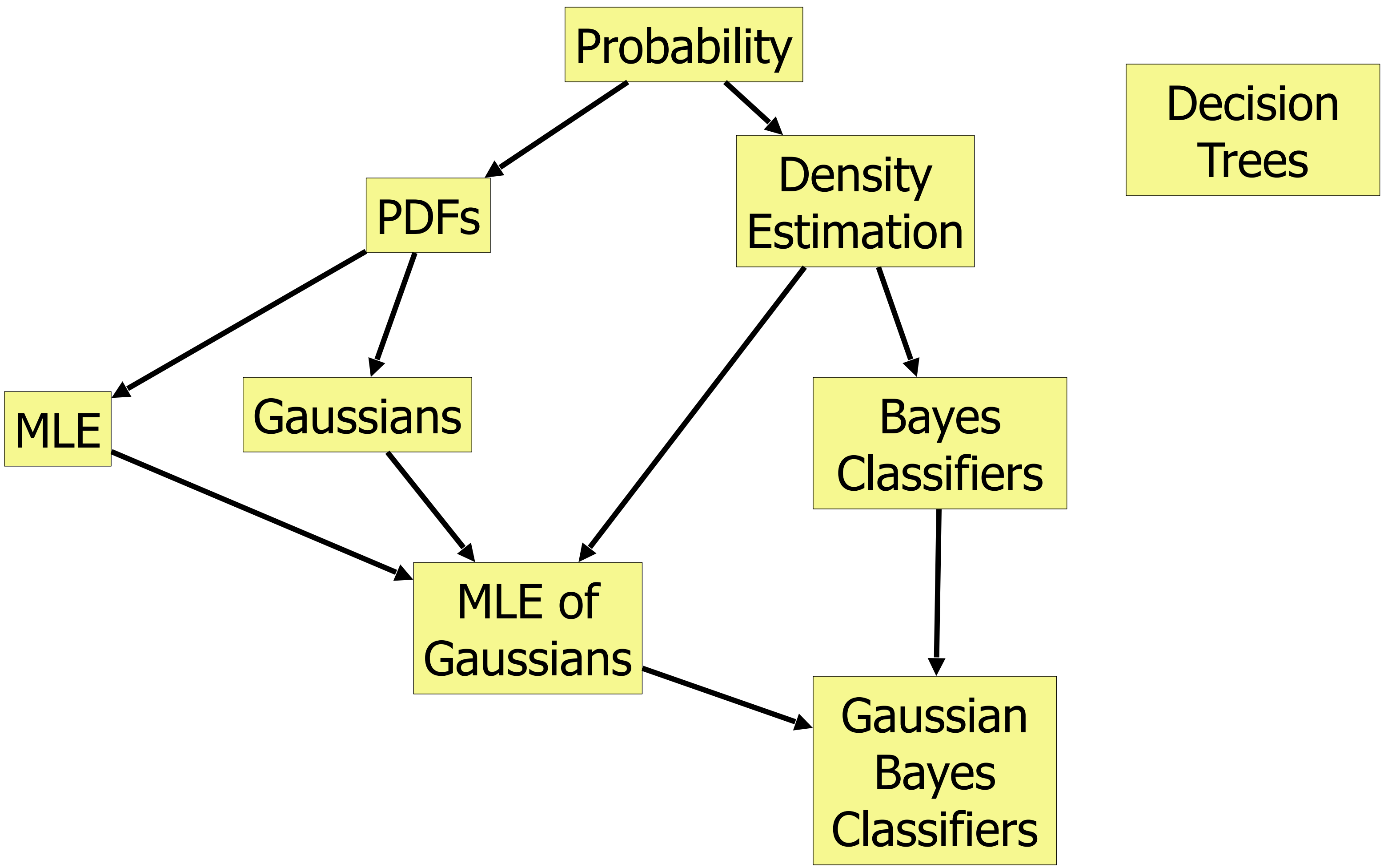
This lecture…

	Categorical inputs only	Real-valued inputs only	Mixed Real / Cat okay
Inputs → Classifier → Predict category	Joint BC Naïve BC	Gauss BC	Dec Tree
Inputs → Density Estimator → Prob-ability	Joint DE Naïve DE	Gauss DE	
Inputs → Regressor → Predict real no.			

Road Map


Probability
Decision Trees
PDFs
Density Estimation
MLE
Gaussians
Bayes Classifiers
MLE of Gaussians

Road Map



Copyright © 2001, Andrew W. Moore

Gaussian Bayes Classifier Assumption

- The i'th record in the database is created using the following algorithm

1. Generate the output (the "class") by drawing $y_i \sim \text{Multinomial}(p_1, p_2, \ldots p_{Ny})$
2. Generate the inputs from a Gaussian PDF that depends on the value of y_i :

$$\mathbf{x}_i \sim N(\mu_i, \Sigma_i).$$

Test your understanding. Given N_y classes and m input attributes, how many distinct scalar parameters need to be estimated?

Copyright © 2001, Andrew W. Moore

MLE Gaussian Bayes Classifier

Let DB_i = Subset of database DB in which the output class is y = i

$$p_i^{mle} = \frac{|DB_i|}{|DB|}$$

the created

g a

1. Generate the output (the "class") by drawing $y_i \sim \text{Multinomial}(p_1, p_2, \ldots p_{N_y})$
2. Generate the inputs from a Gaussian PDF that depends on the value of y_i :

$$\mathbf{x}_i \sim N(\mu_i, \Sigma_i).$$

Test your understanding. Given N_y classes and m input attributes, how many distinct scalar parameters need to be estimated?

Copyright © 2001, Andrew W. Moore

MLE Gaussian Bayes Classifier

Let DB_i = Subset of database DB in which the output class is $y = i$

the database is created
g algorithm

1. (Generate the output (the "class") by

$(\mu_i^{mle}, \Sigma_i^{mle})$= MLE Gaussian for DB_i

2. Generate the inputs fr aussian PDF that depends on the va e of y_i :

$$\mathbf{x}_i \sim N(\mu_i , \Sigma_i).$$

Test your understanding. Given N_y classes and m input attributes, how many distinct scalar parameters need to be estimated?

Copyright © 2001, Andrew W. Moore

MLE Gaussian Bayes Classifier

Let DB_i = Subset of database DB in which the output class is $y = i$

the database is created
g algorithm

1. … wing

$(\mu_i^{mle}, \Sigma_i^{mle})$ = MLE Gaussian for DB_i

2. Generate the inputs fr… aussian PDF that depends on the va… e of y_i :

$$\mathbf{x}_i \sim N(\mu_i, \Sigma_i).$$

$$\boldsymbol{\mu}_i^{mle} = \frac{1}{|\mathrm{DB}_i|} \sum_{\mathbf{x}_k \in \mathrm{DB}_i}^{R} \mathbf{x}_k$$

$$\boldsymbol{\Sigma}_i^{mle} = \frac{1}{|\mathrm{DB}_i|} \sum_{\mathbf{x}_k \in \mathrm{DB}_i}^{R} \left(\mathbf{x}_k - \boldsymbol{\mu}_i^{mle}\right)\left(\mathbf{x}_k - \boldsymbol{\mu}_i^{mle}\right)^T$$

g. Gi… y distinct scalar parameters need to be estimated?

Copyright © 2001, Andrew W. Moore

Gaussian Bayes Classification

$$P(y=i \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid y=i)P(y=i)}{p(\mathbf{x})}$$

Gaussian Bayes Classification

$$P(y=i \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid y=i)P(y=i)}{p(\mathbf{x})}$$

$$P(y=i \mid \mathbf{x}) = \frac{\frac{1}{(2\pi)^{m/2} \|\, \boldsymbol{\Sigma}_i \,\|^{1/2}} \exp\left[-\frac{1}{2}\left(\mathbf{x}_k - \boldsymbol{\mu}_i\right)^T \boldsymbol{\Sigma}_i \left(\mathbf{x}_k - \boldsymbol{\mu}_i\right)\right] p_i}{p(\mathbf{x})}$$

How do we deal with that?

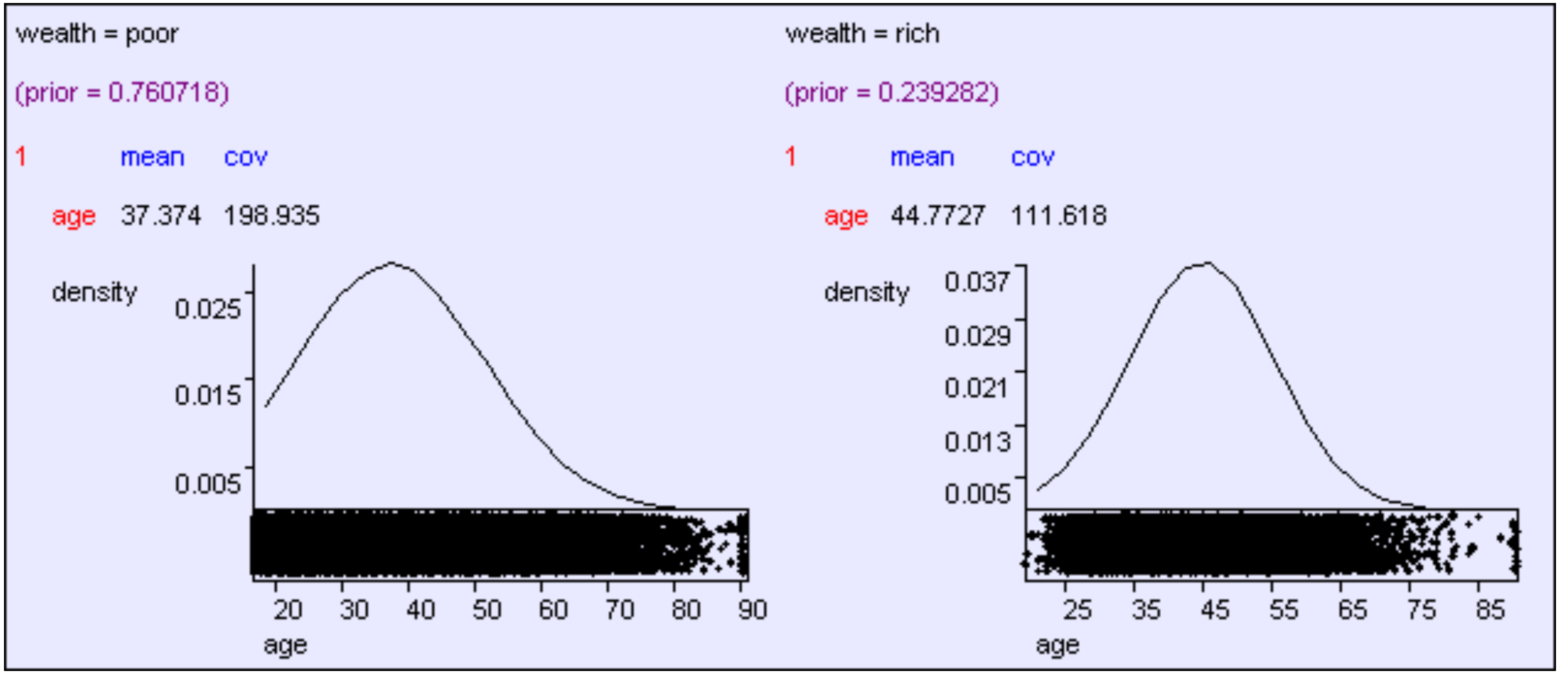
Here is a dataset

age	employme	education	edun	marital	…	job	relation	race	gender	hour	country	wealth
					…							
39	State_gov	Bachelors	13	Never_mar	…	Adm_cleric	Not_in_fam	White	Male	40	United_Sta	poor
51	Self_emp_	Bachelors	13	Married	…	Exec_man	Husband	White	Male	13	United_Sta	poor
39	Private	HS_grad	9	Divorced	…	Handlers_c	Not_in_fam	White	Male	40	United_Sta	poor
54	Private	11th	7	Married	…	Handlers_c	Husband	Black	Male	40	United_Sta	poor
28	Private	Bachelors	13	Married	…	Prof_speci	Wife	Black	Female	40	Cuba	poor
38	Private	Masters	14	Married	…	Exec_man	Wife	White	Female	40	United_Sta	poor
50	Private	9th	5	Married_sp	…	Other_serv	Not_in_fam	Black	Female	16	Jamaica	poor
52	Self_emp_	HS_grad	9	Married	…	Exec_man	Husband	White	Male	45	United_Sta	rich
31	Private	Masters	14	Never_mar	…	Prof_speci	Not_in_fam	White	Female	50	United_Sta	rich
42	Private	Bachelors	13	Married	…	Exec_man	Husband	White	Male	40	United_Sta	rich
37	Private	Some_coll	10	Married	…	Exec_man	Husband	Black	Male	80	United_Sta	rich
30	State_gov	Bachelors	13	Married	…	Prof_speci	Husband	Asian	Male	40	India	rich
24	Private	Bachelors	13	Never_mar	…	Adm_cleric	Own_child	White	Female	30	United_Sta	poor
33	Private	Assoc_acc	12	Never_mar	…	Sales	Not_in_fam	Black	Male	50	United_Sta	poor
41	Private	Assoc_voc	11	Married	…	Craft_repai	Husband	Asian	Male	40	*MissingVa	rich
34	Private	7th_8th	4	Married	…	Transport_	Husband	Amer_India	Male	45	Mexico	poor
26	Self_emp_	HS_grad	9	Never_mar	…	Farming_fi	Own_child	White	Male	35	United_Sta	poor
33	Private	HS_grad	9	Never_mar	…	Machine_o	Unmarried	White	Male	40	United_Sta	poor
38	Private	11th	7	Married	…	Sales	Husband	White	Male	50	United_Sta	poor
44	Self_emp_	Masters	14	Divorced	…	Exec_man	Unmarried	White	Female	45	United_Sta	rich
41	Private	Doctorate	16	Married	…	Prof_speci	Husband	White	Male	60	United_Sta	rich
:	:	:	:	:	:	:	:	:	:	:	:	:

48,000 records, 16 attributes [Kohavi 1995]

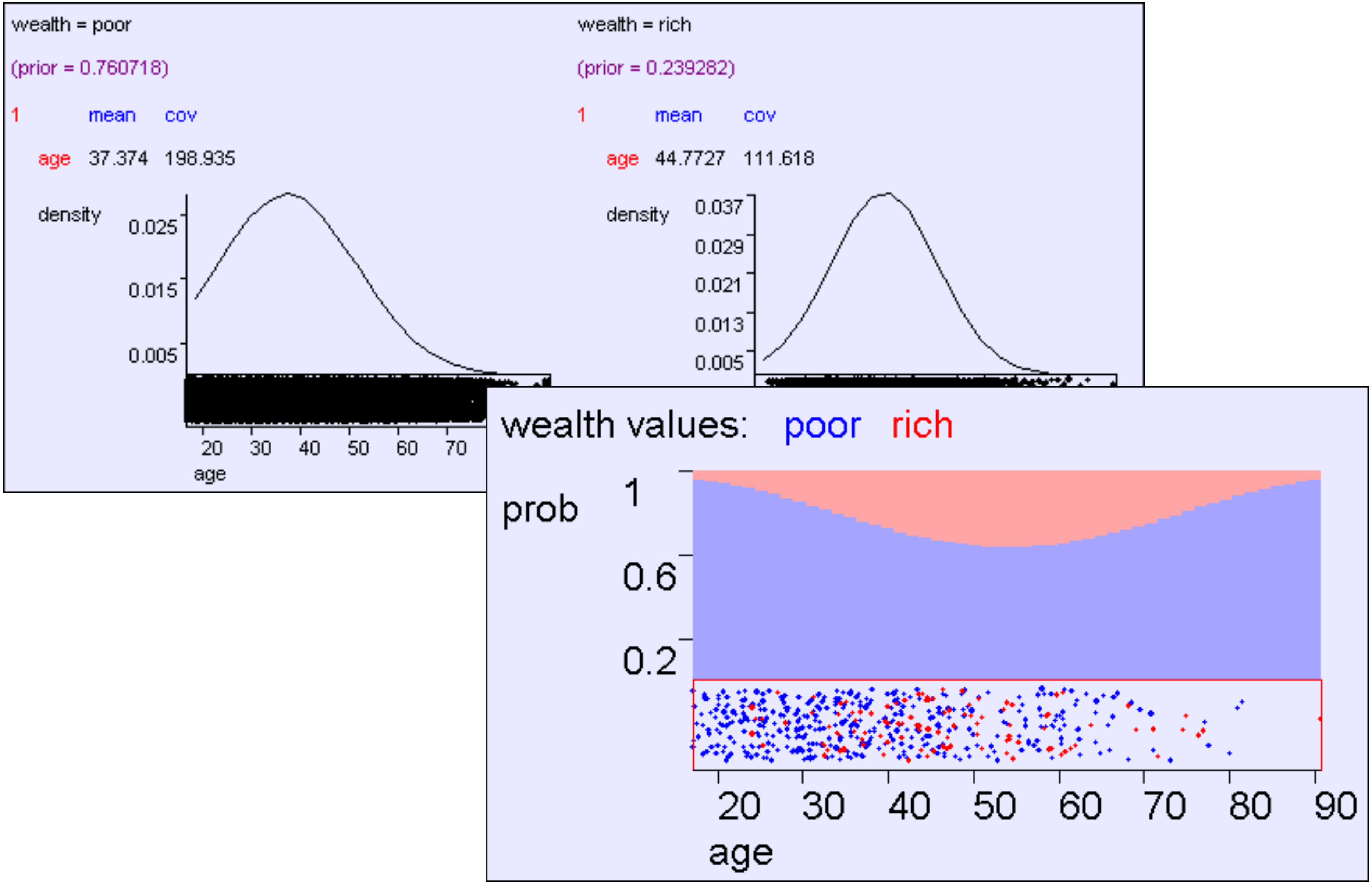
Copyright © 2001, Andrew W. Moore

Predicting wealth from age

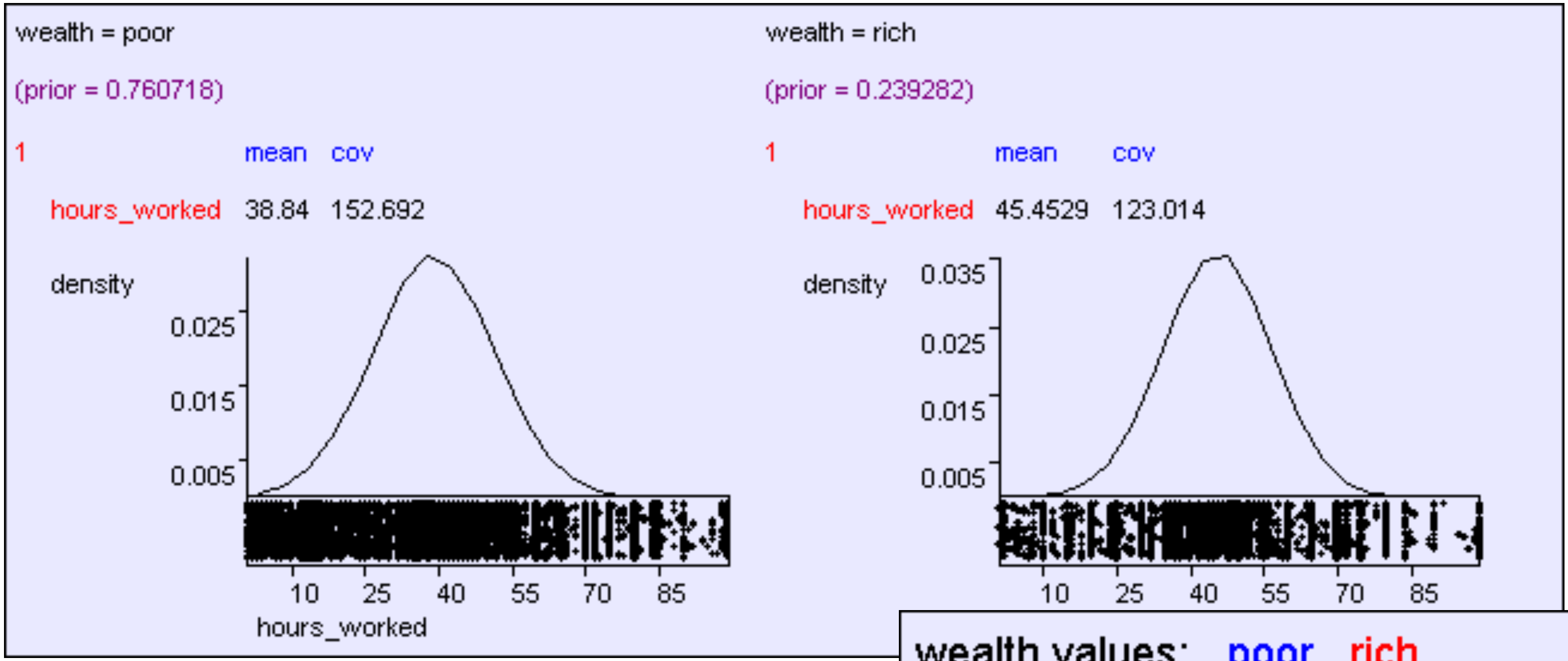


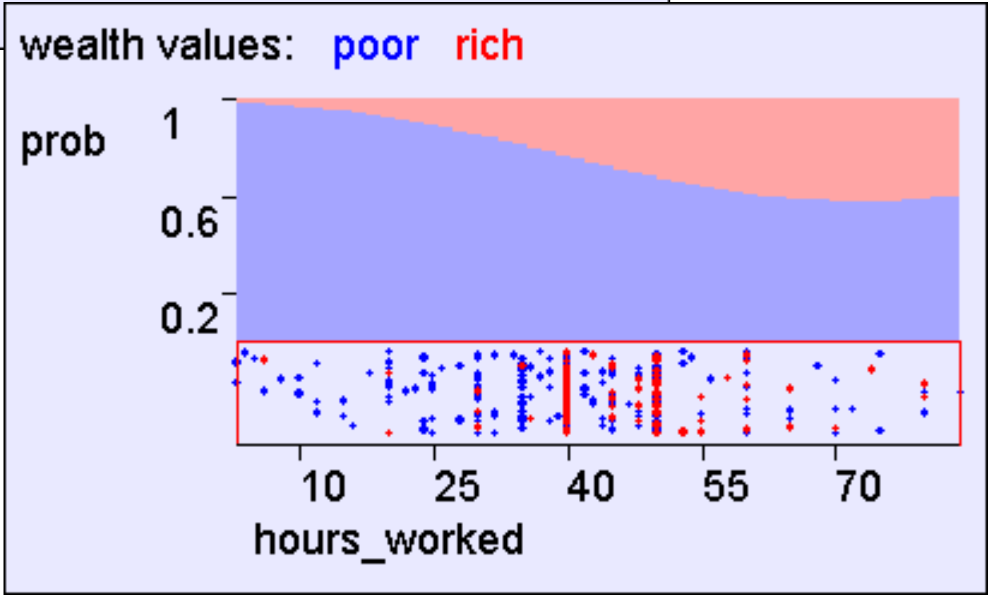
Copyright © 2001, Andrew W. Moore

Predicting wealth from age



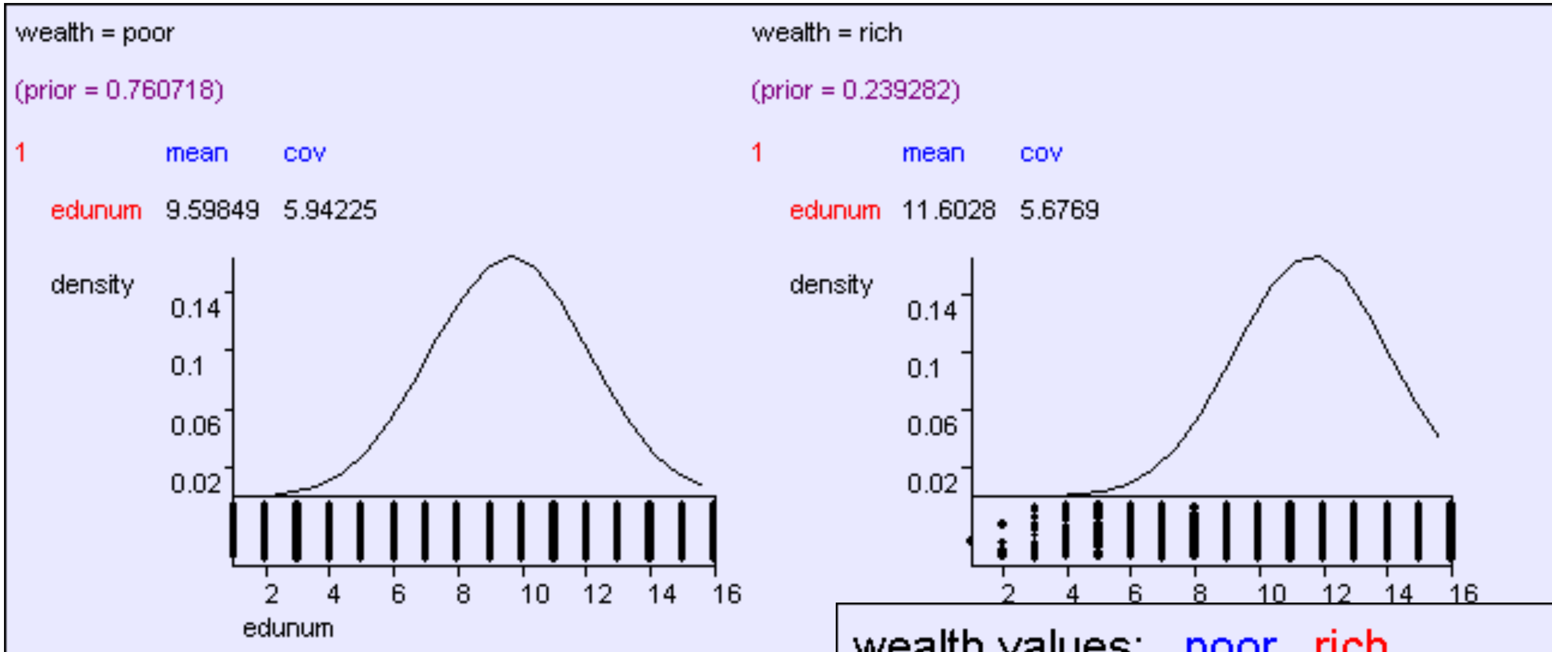
Wealth from hours worked

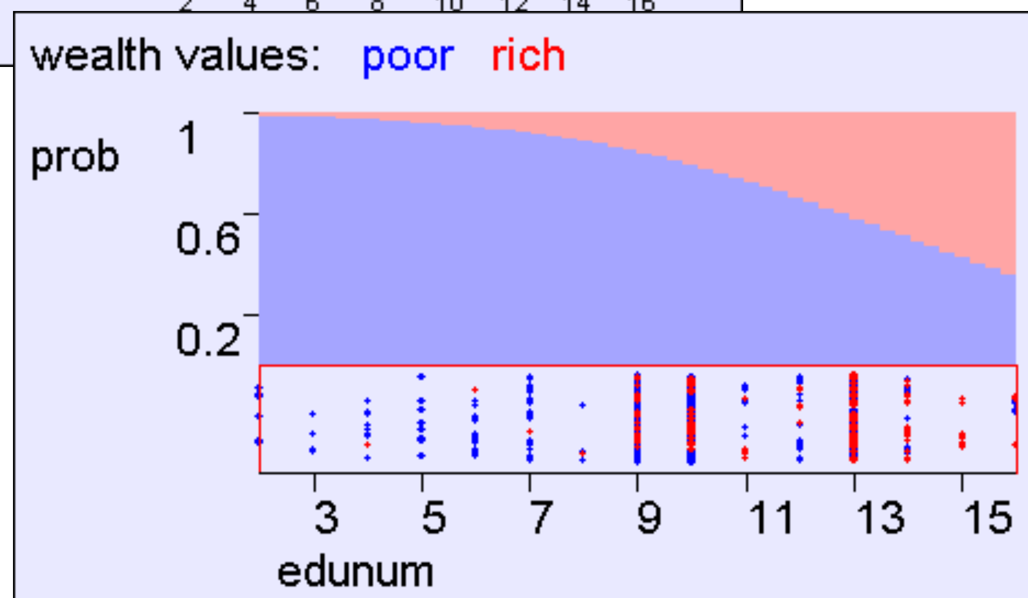




Copyright © 2001, Andrew W. Moore

Wealth from years of education



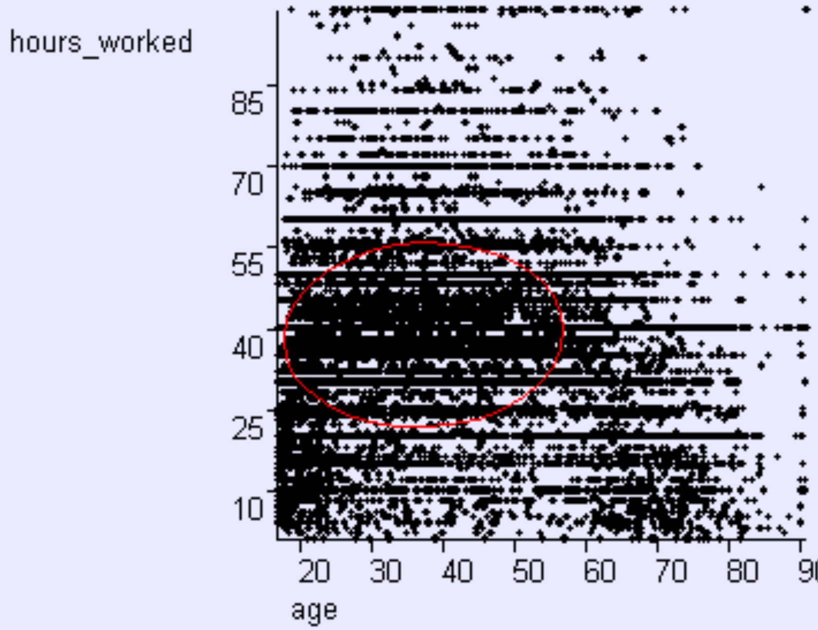


age, hours → wealth

wealth = poor

(prior = 0.760718)

1	mean	cov	
age	37.374	198.935	8.70283
hours_worked	38.84	8.70283	152.692



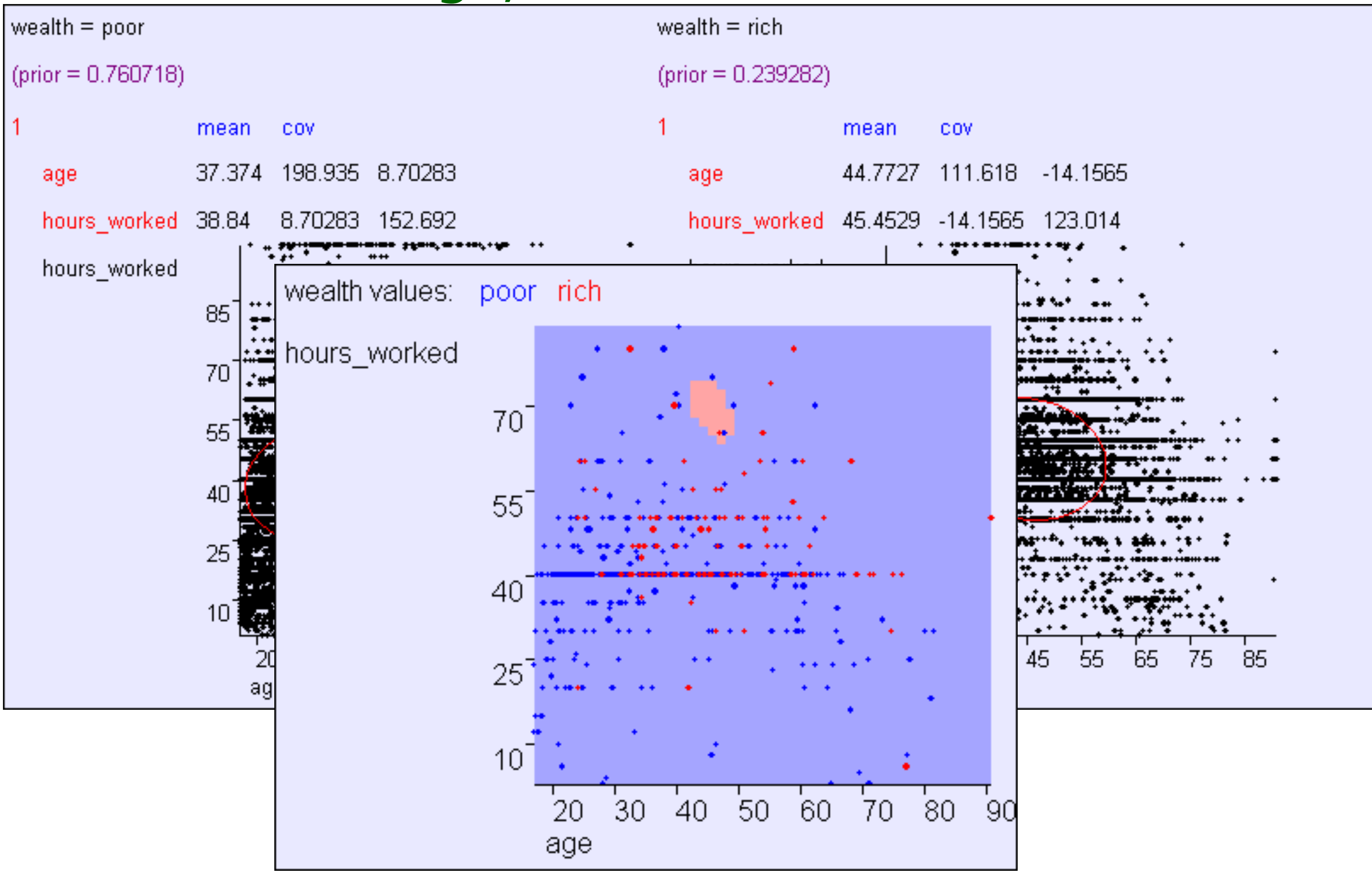
wealth = rich

(prior = 0.239282)

1	mean	cov	
age	44.7727	111.618	-14.1565
hours_worked	45.4529	-14.1565	123.014

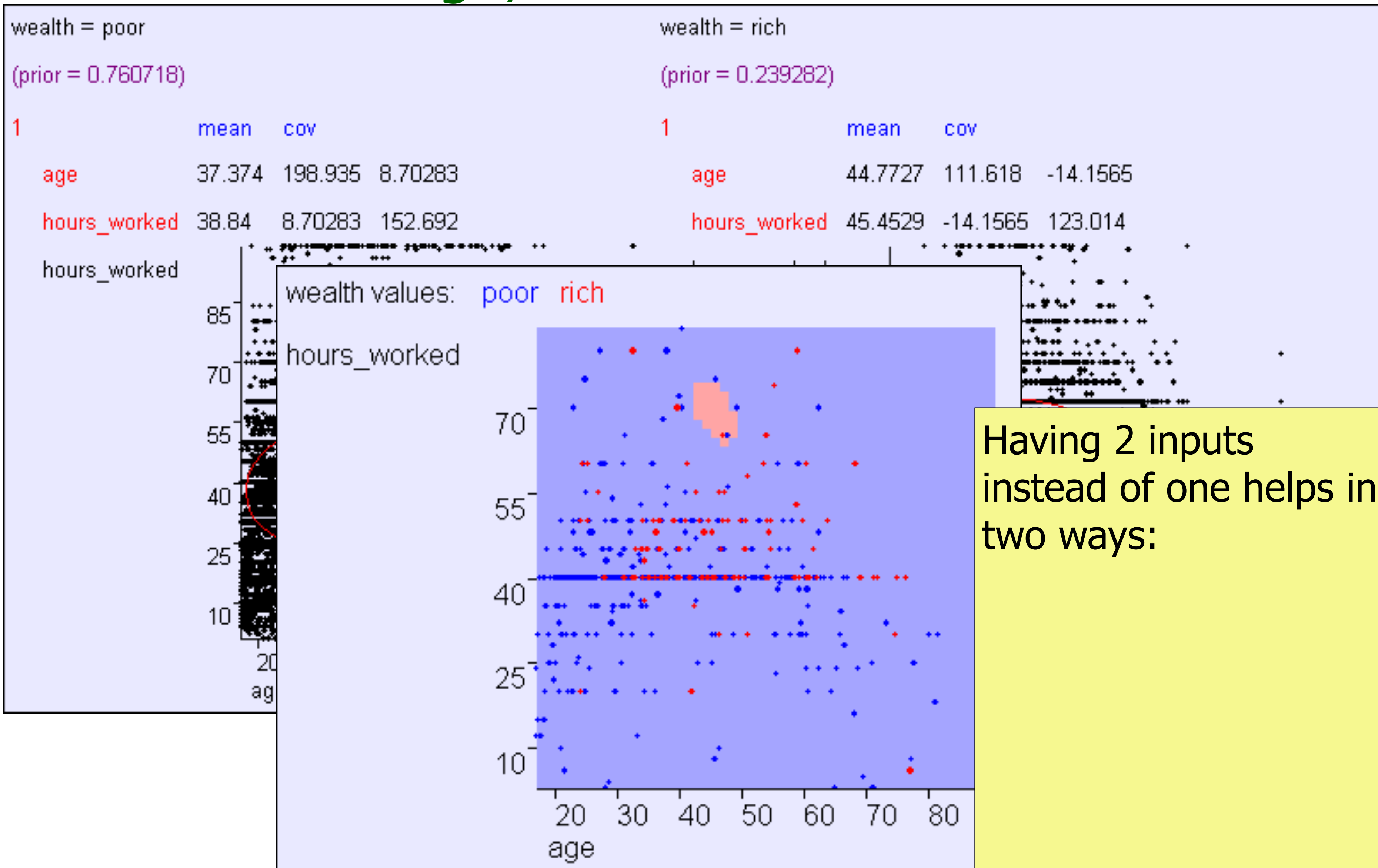
Copyright © 2001, Andrew W. Moore

age, hours → wealth



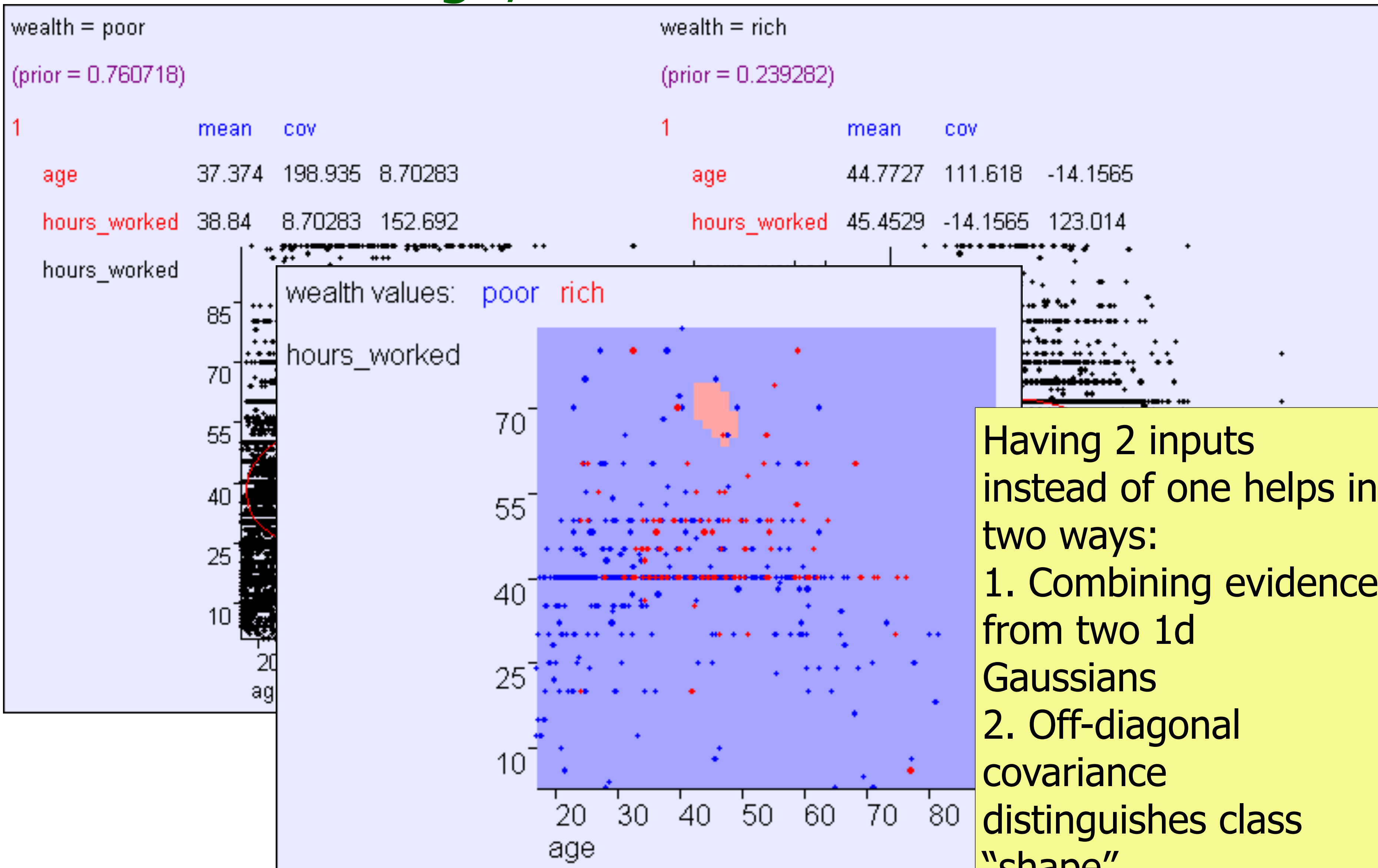
Copyright © 2001, Andrew W. Moore

age, hours → wealth



Having 2 inputs instead of one helps in two ways:

age, hours → wealth

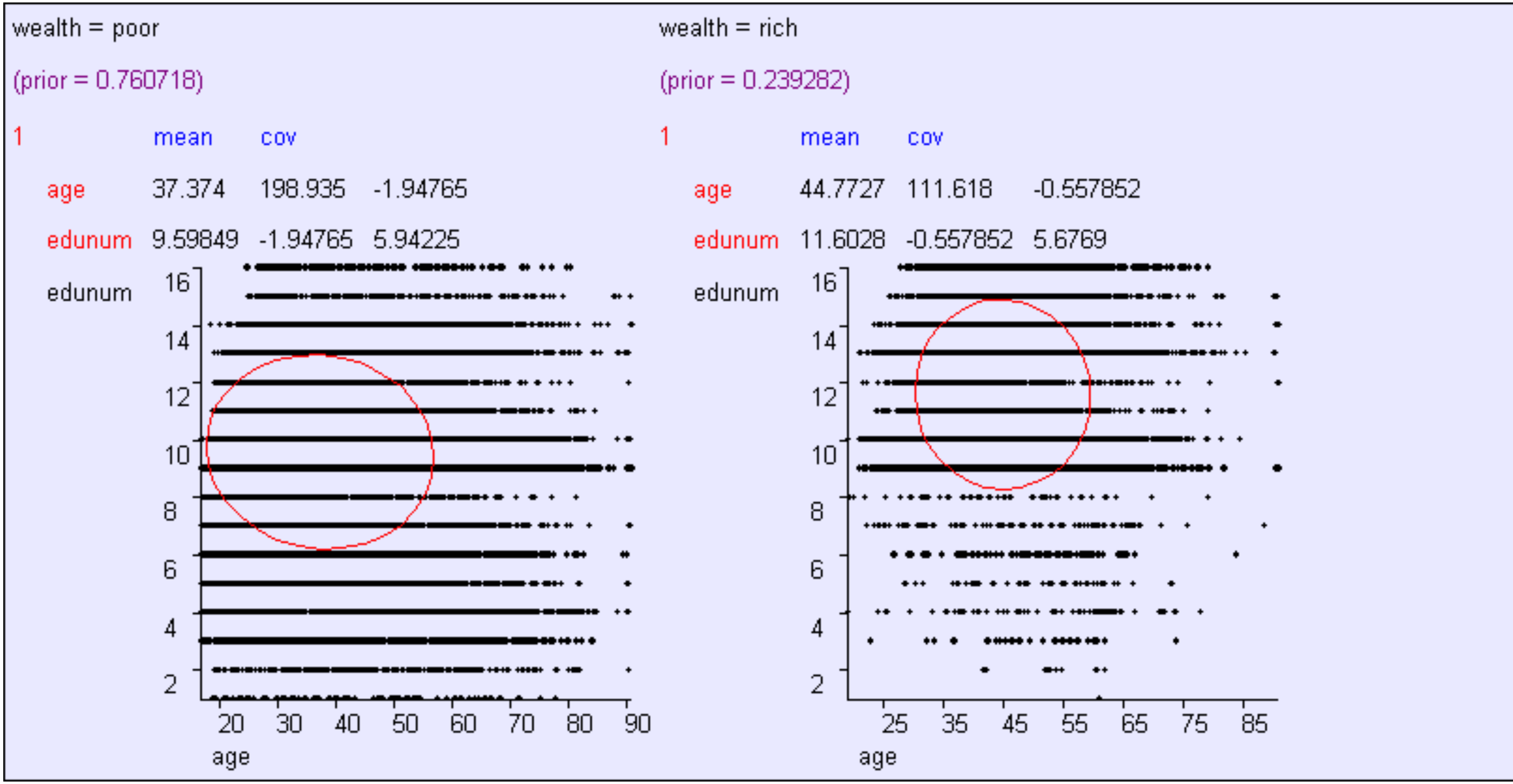


Having 2 inputs instead of one helps in two ways:

1. Combining evidence from two 1d Gaussians
2. Off-diagonal covariance distinguishes class "shape"

Copyright © 2001, Andrew W. Moore

age, edunum → wealth



age, edunum → wealth

wealth = poor

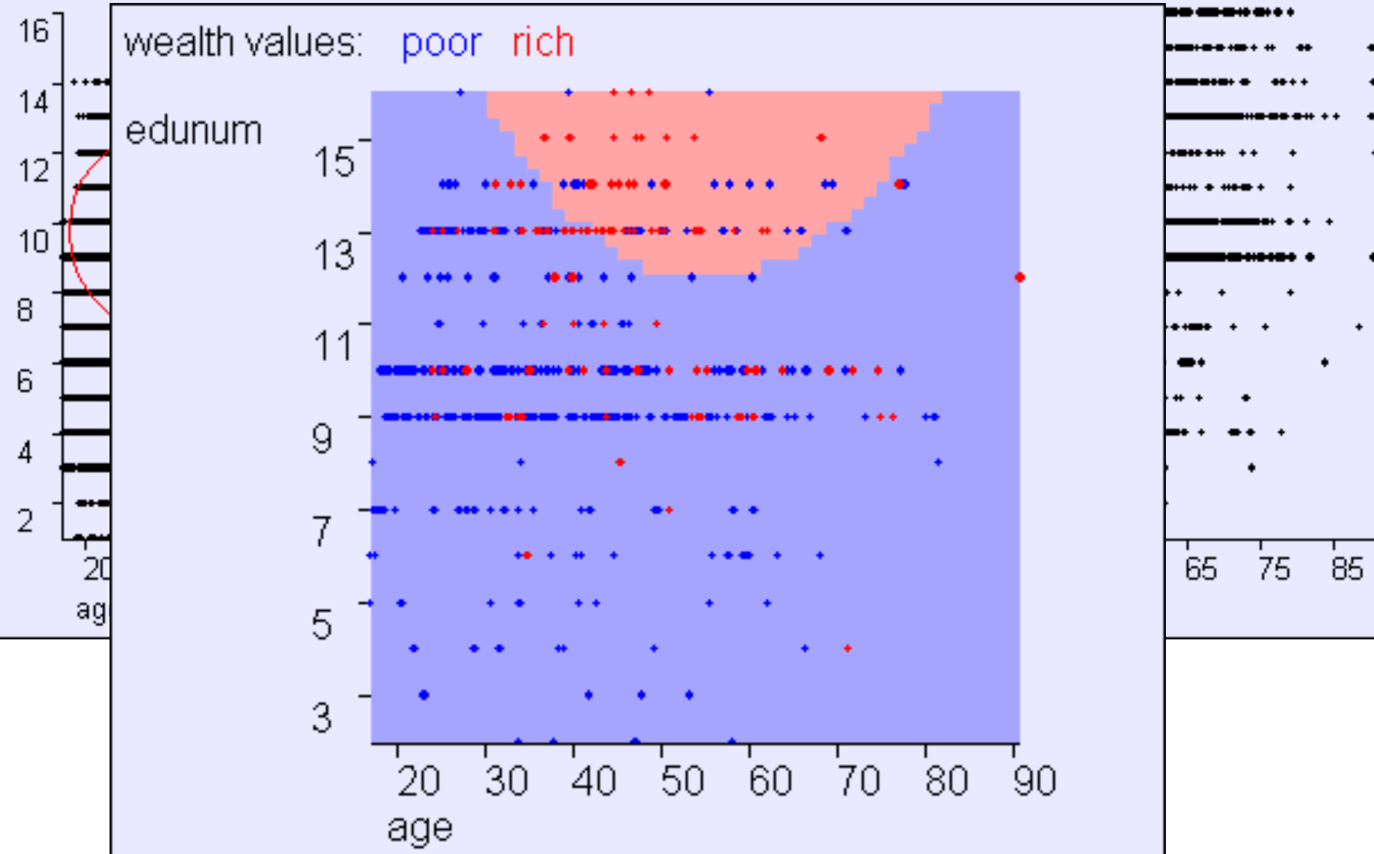
(prior = 0.760718)

1	mean	cov	
age	37.374	198.935	-1.94765
edunum	9.59849	-1.94765	5.94225

wealth = rich

(prior = 0.239282)

1	mean	cov	
age	44.7727	111.618	-0.557852
edunum	11.6028	-0.557852	5.6769

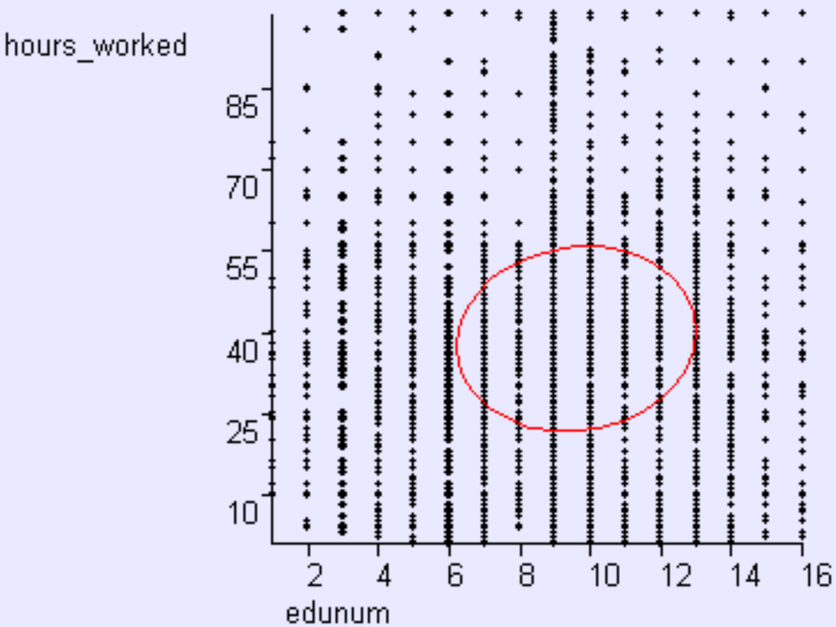


hours, edunum → wealth

wealth = poor

(prior = 0.760718)

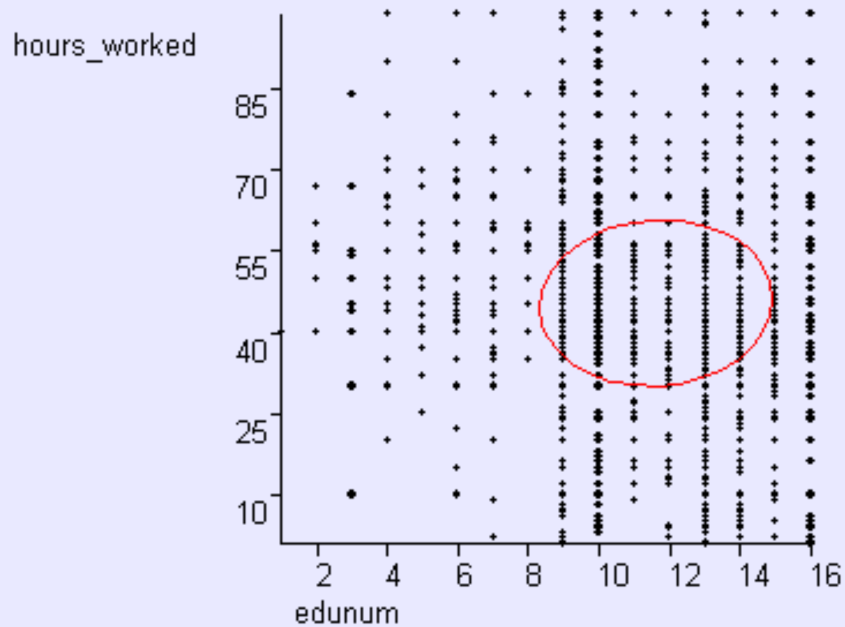
1	mean	cov	
edunum	9.59849	5.94225	2.4298
hours_worked	38.84	2.4298	152.692



wealth = rich

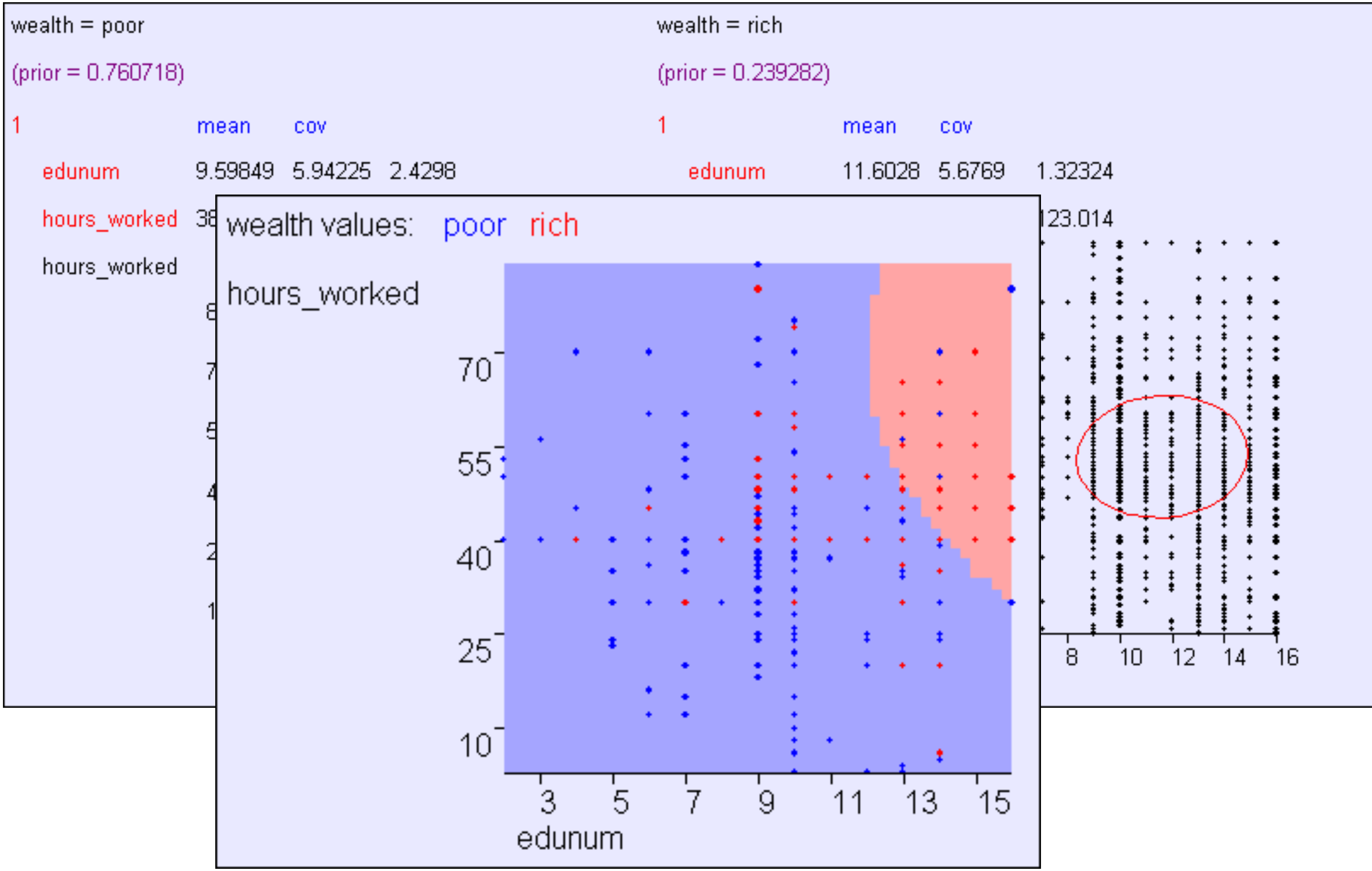
(prior = 0.239282)

1	mean	cov	
edunum	11.6028	5.6769	1.32324
hours_worked	45.4529	1.32324	123.014



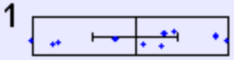
Copyright © 2001, Andrew W. Moore

hours, edunum → wealth

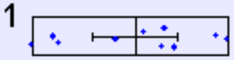


Accuracy

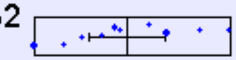
Name	Model	Parameters	FracRight
age+hours	bayesclass	density=joint submodel=gauss gausstype=general	0.760452 +/- 0.00319521



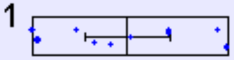
Name	Model	Parameters	FracRight
age+hours	bayesclass	density=joint submodel=gauss gausstype=general	0.760452 +/- 0.00319521



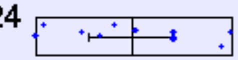
Name	Model	Parameters	FracRight
age+hours+edunum	bayesclass	density=joint submodel=gauss gausstype=general	0.798513 +/- 0.00542432



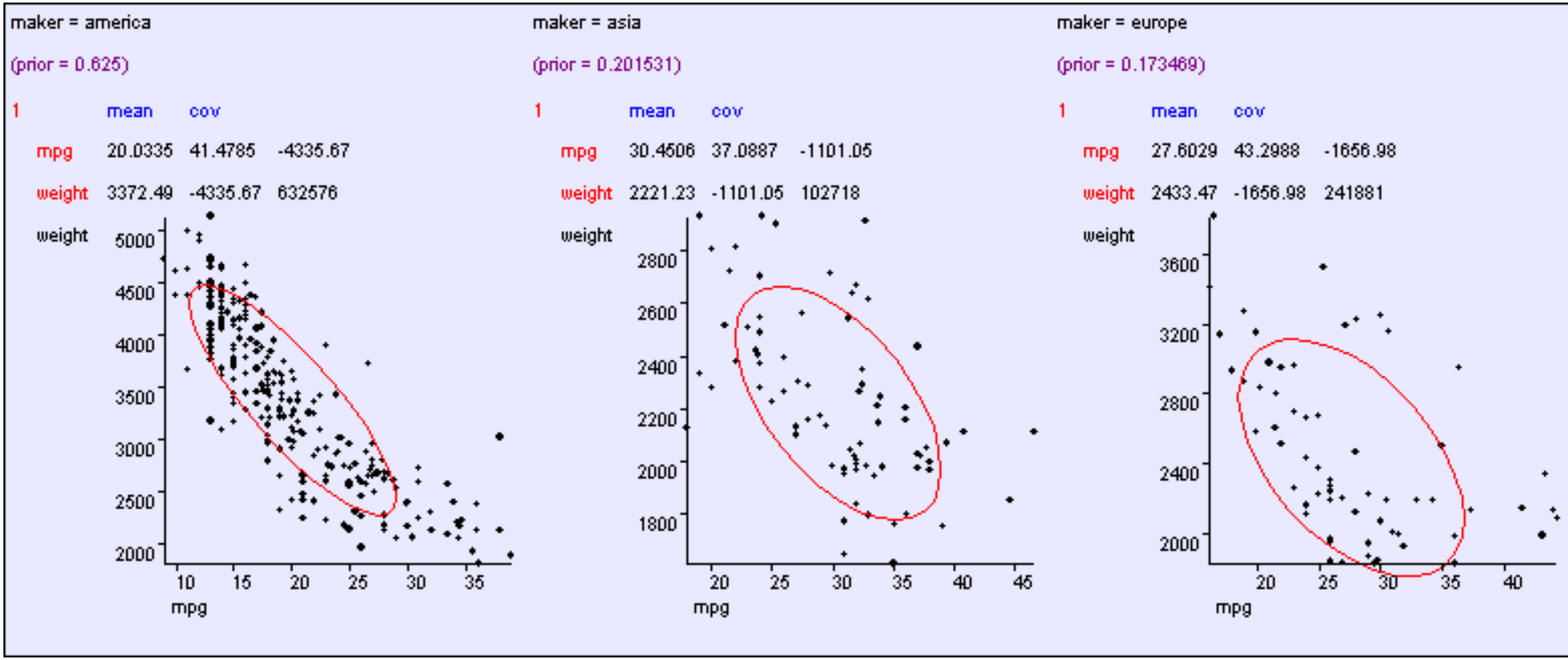
Name	Model	Parameters	FracRight
a+h+e+capgain	bayesclass	density=joint submodel=gauss gausstype=general	0.793518 +/- 0.00319241



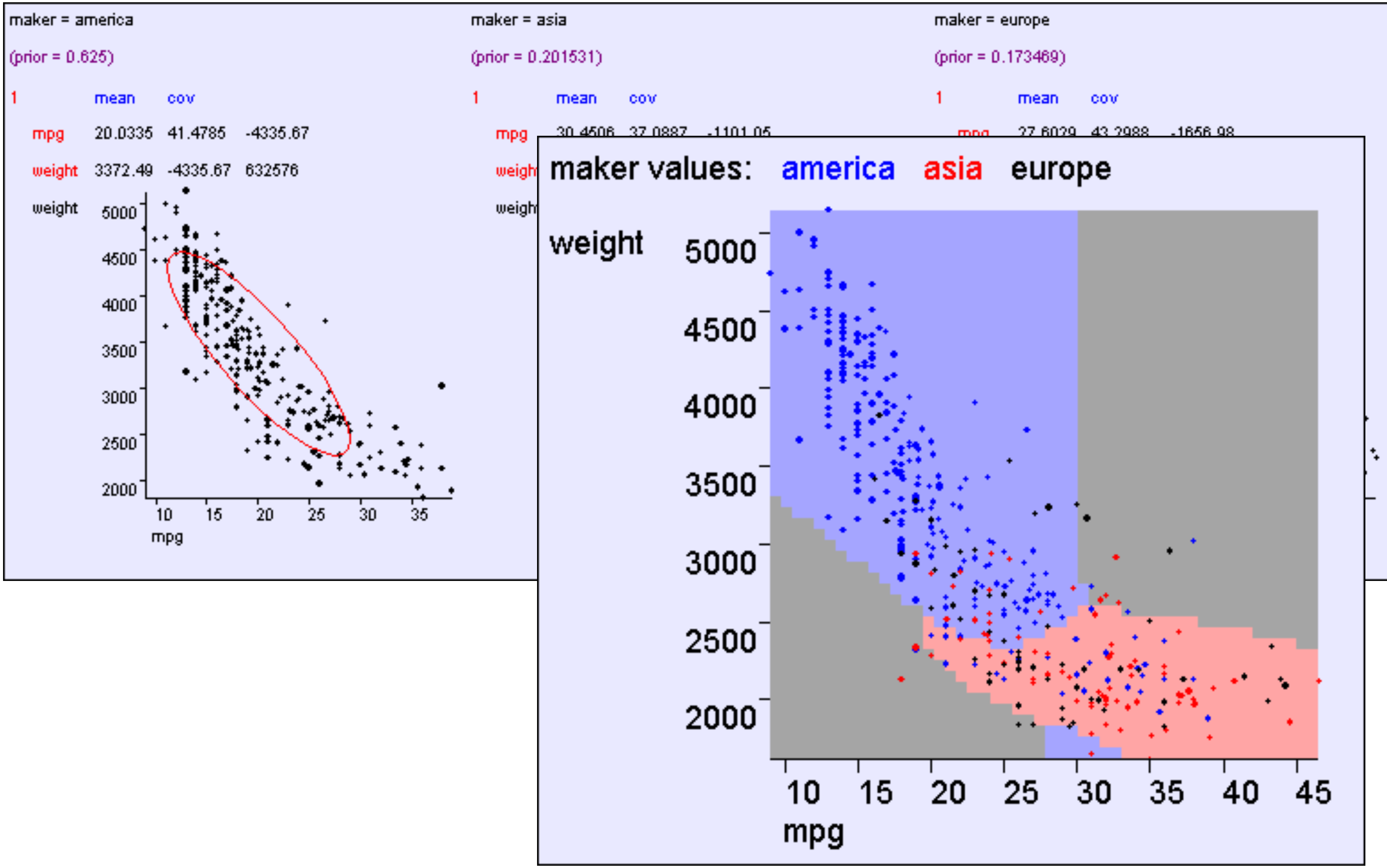
Name	Model	Parameters	FracRight
a+h+e+c+taxweight	bayesclass	density=joint submodel=gauss gausstype=general	0.793477 +/- 0.00321524



An “MPG” example



An "MPG" example



Copyright © 2001, Andrew W. Moore

An "MPG" example

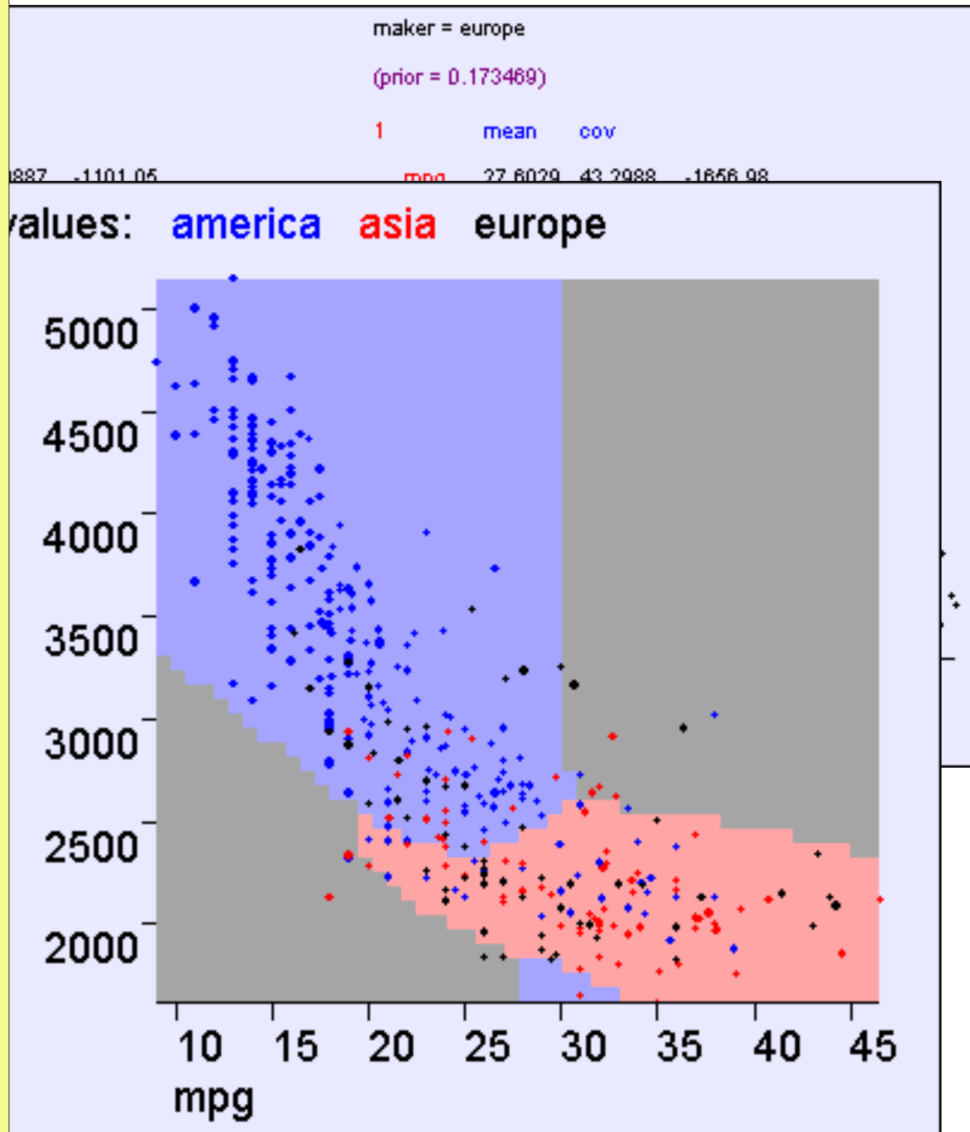
Things to note:

•Class Boundaries can be weird shapes (hyperconic sections)

•Class regions can be non-simply-connected

•But it's impossible to model arbitrarily weirdly shaped regions

•**Test your understanding:** With one input, must classes be simply connected?



Overfitting dangers

- Problem with “Joint” Bayes classifier:

  #parameters exponential with #dimensions.

  This means we just memorize the training data, and can overfit.

Copyright © 2001, Andrew W. Moore

Overfitting dangers

- Problem with "Joint" Bayes classifier:

  #parameters exponential with #dimensions.

  This means we just memorize the training data, and can overfit.

- Problemette with Gaussian Bayes classifier:

  #parameters quadratic with #dimensions.

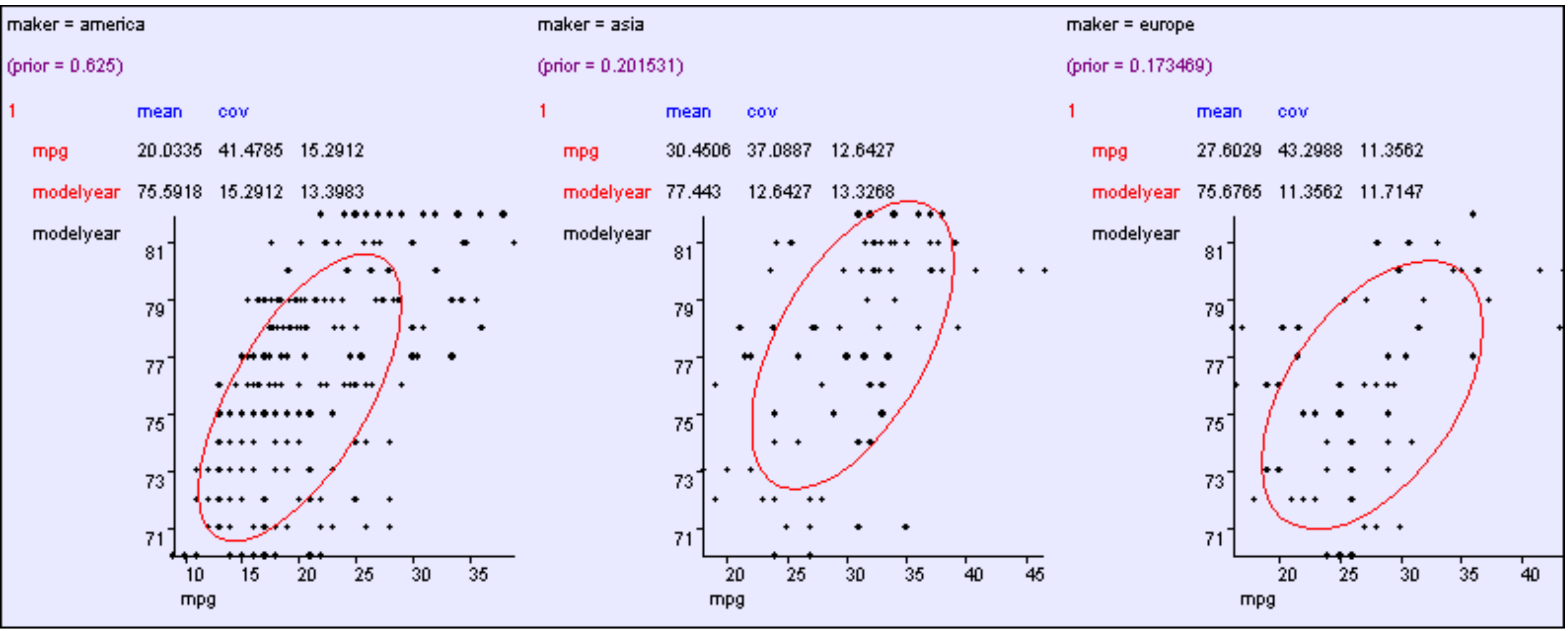
  With 10,000 dimensions and only 1,000 datapoints we could overfit.

Question: Any suggested solutions?

Copyright © 2001, Andrew W. Moore

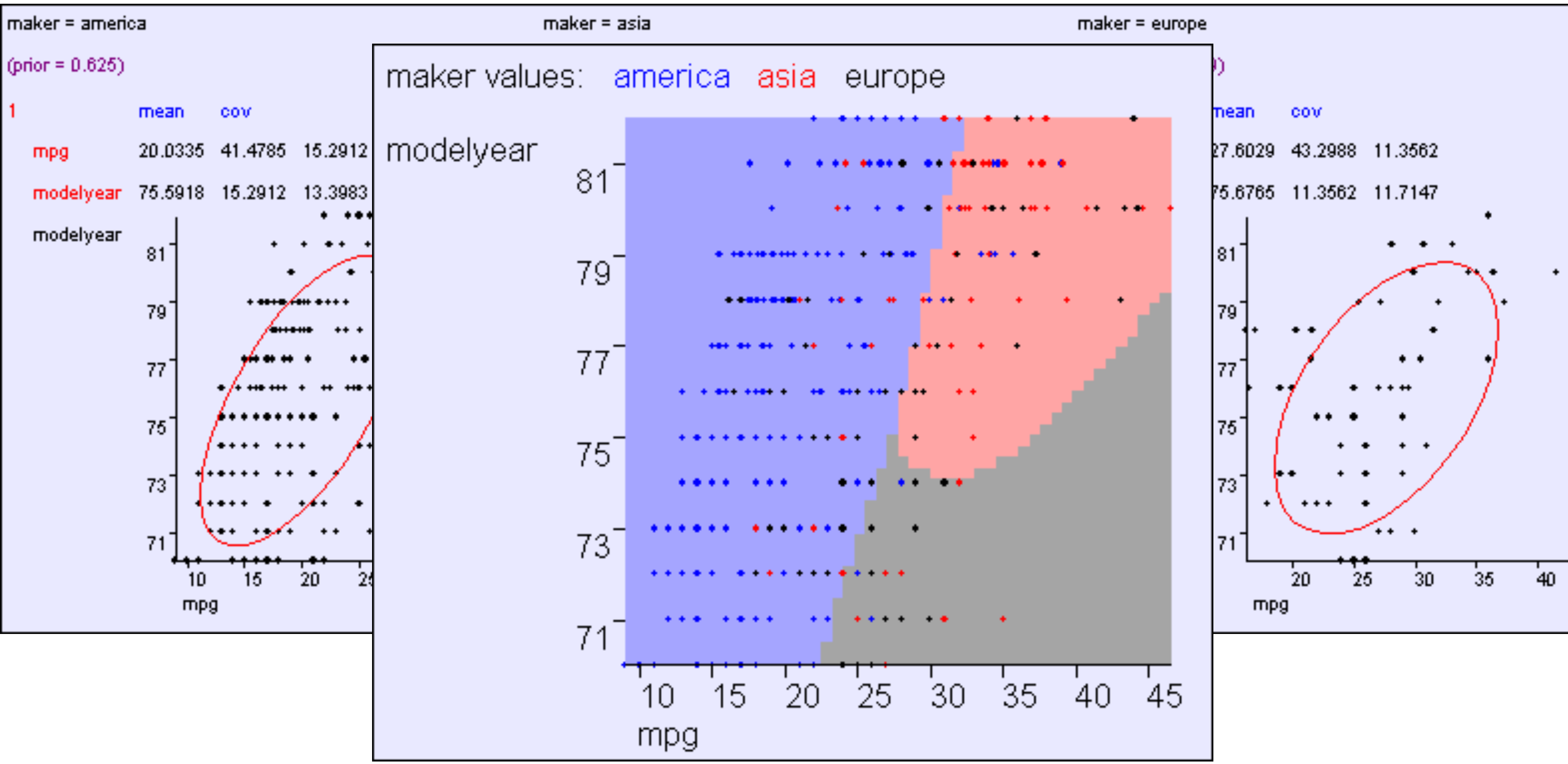
General: O(m²) parameters

$$\Sigma = \begin{pmatrix} \sigma^2{}_1 & \sigma_{12} & \boxed{?} & \sigma_{1m} \\ \sigma_{12} & \sigma^2{}_2 & \boxed{?} & \sigma_{2m} \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \sigma_{1m} & \sigma_{2m} & \boxed{?} & \sigma^2{}_m \end{pmatrix}$$



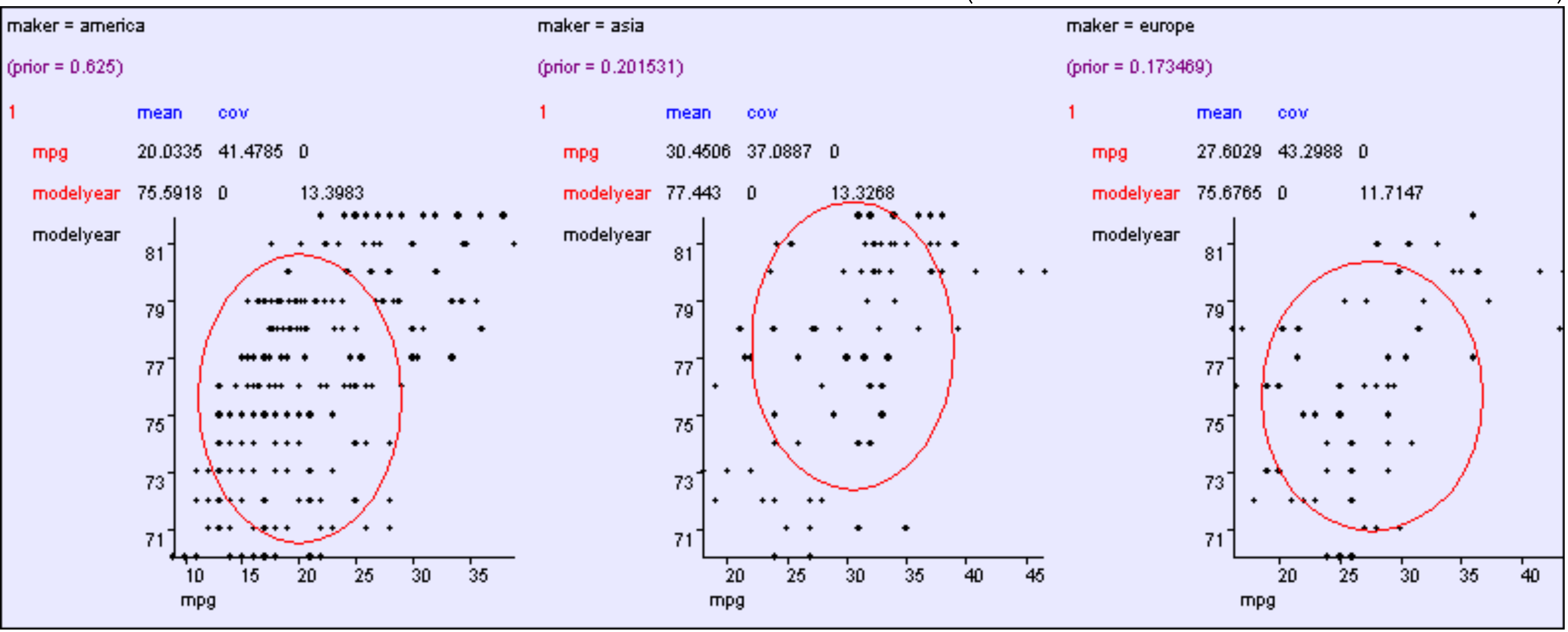
General: O(m²) parameters

$$\Sigma = \begin{pmatrix} \sigma^2_1 & \sigma_{12} & \boxed{?} & \sigma_{1m} \\ \sigma_{12} & \sigma^2_2 & \boxed{?} & \sigma_{2m} \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ \sigma_{1m} & \sigma_{2m} & \boxed{?} & \sigma^2_m \end{pmatrix}$$



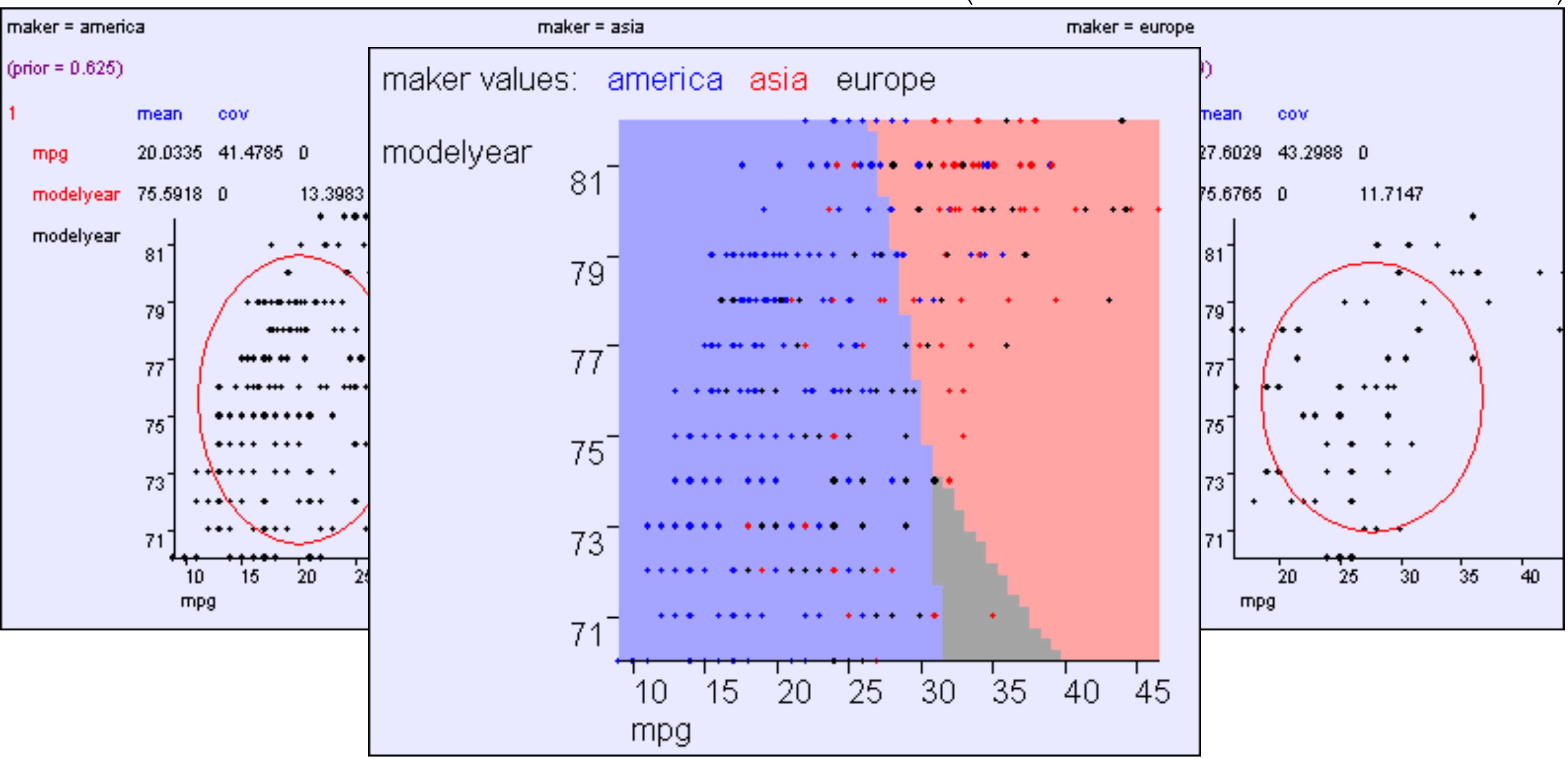
Aligned: O(m) parameters

$$\Sigma = \begin{pmatrix} \sigma^2{}_1 & 0 & 0 & \boxed{?} & 0 & 0 \\ 0 & \sigma^2{}_2 & 0 & \boxed{?} & 0 & 0 \\ 0 & 0 & \sigma^2{}_3 & \boxed{?} & 0 & 0 \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ 0 & 0 & 0 & \boxed{?} & \sigma^2{}_{m-1} & 0 \\ 0 & 0 & 0 & \boxed{?} & 0 & \sigma^2{}_m \end{pmatrix}$$



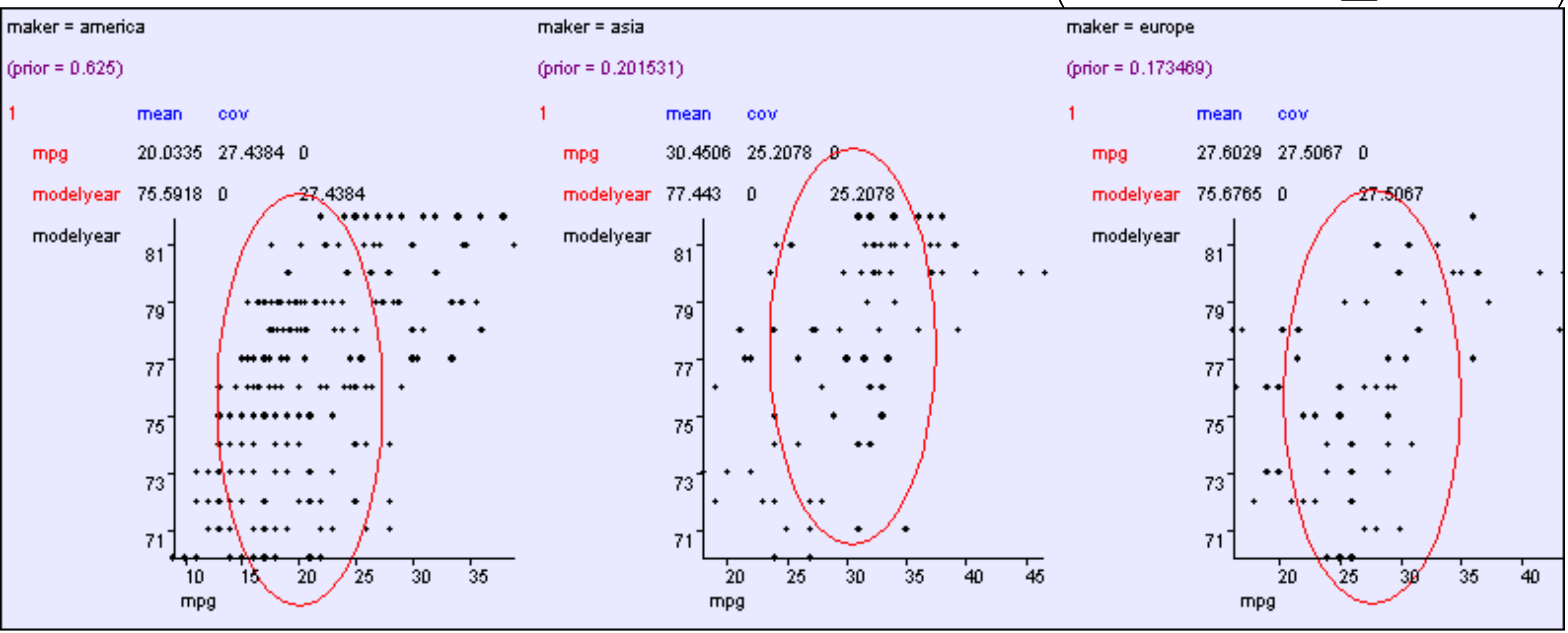
Aligned: O(m) parameters

$$\Sigma = \begin{pmatrix} \sigma^2_1 & 0 & 0 & \boxed{?} & 0 & 0 \\ 0 & \sigma^2_2 & 0 & \boxed{?} & 0 & 0 \\ 0 & 0 & \sigma^2_3 & \boxed{?} & 0 & 0 \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ 0 & 0 & 0 & \boxed{?} & \sigma^2_{m-1} & 0 \\ 0 & 0 & 0 & \boxed{?} & 0 & \sigma^2_m \end{pmatrix}$$



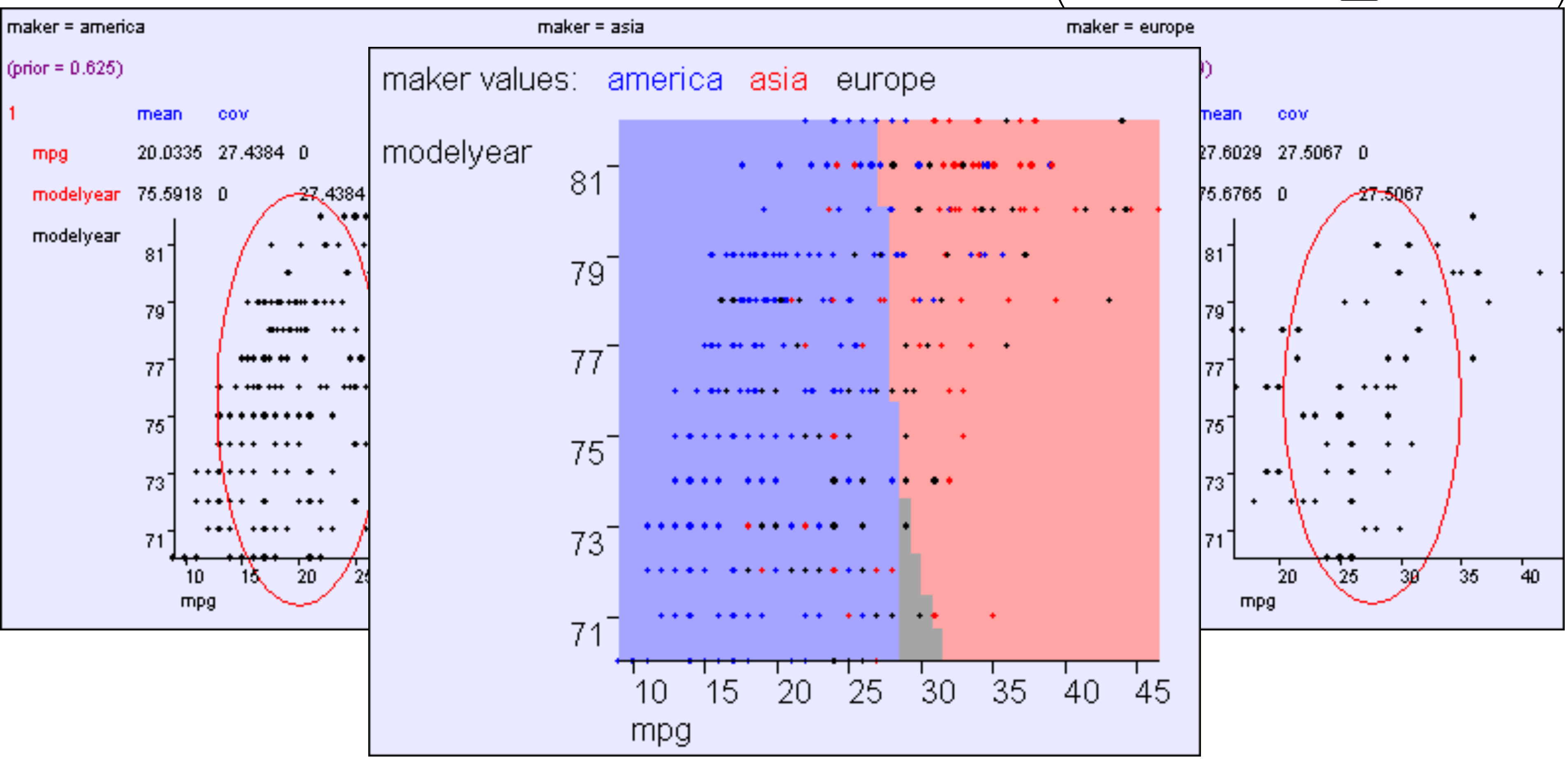
Spherical: O(1) cov parameters

$$\Sigma = \begin{pmatrix} \sigma^2 & 0 & 0 & \boxed{?} & 0 & 0 \\ 0 & \sigma^2 & 0 & \boxed{?} & 0 & 0 \\ 0 & 0 & \sigma^2 & \boxed{?} & 0 & 0 \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ 0 & 0 & 0 & \boxed{?} & \sigma^2 & 0 \\ 0 & 0 & 0 & \boxed{?} & 0 & \sigma^2 \end{pmatrix}$$



Spherical: O(1) cov parameters

$$\Sigma = \begin{pmatrix} \sigma^2 & 0 & 0 & \boxed{?} & 0 & 0 \\ 0 & \sigma^2 & 0 & \boxed{?} & 0 & 0 \\ 0 & 0 & \sigma^2 & \boxed{?} & 0 & 0 \\ \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} & \boxed{?} \\ 0 & 0 & 0 & \boxed{?} & \sigma^2 & 0 \\ 0 & 0 & 0 & \boxed{?} & 0 & \sigma^2 \end{pmatrix}$$



BCs that have both real and categorical inputs?

	Categorical inputs only	Real-valued inputs only	Mixed Real / Cat okay
Inputs Classifier Predict category	Joint BC Naïve BC	Gauss BC	Dec Tree BC Here???
Inputs Density Estimator Prob-ability	Joint DE Naïve DE	Gauss DE	
Inputs Regressor Predict real no.			

Copyright © 2001, Andrew W. Moore

BCs that have both real and categorical inputs?

	Categorical inputs only	Real-valued inputs only	Mixed Real / Cat okay
Inputs → Classifier → Predict category	Joint BC Naïve BC	Gauss BC	Dec Tree BC Here???
Inputs → Density Estimator → Prob-ability	Joint DE Naïve DE	Gauss DE	
Inputs → Regressor → Predict real no.			

Easy!

Guess how?

BCs that have both real and categorical inputs?

	Categorical inputs only	Real-valued inputs only	Mixed Real / Cat okay
Inputs → Classifier → Predict category	Joint BC Naïve BC	Gauss BC	Dec Tree Gauss/Joint BC Gauss Naïve BC
Inputs → Density Estimator → Prob-ability	Joint DE Naïve DE	Gauss DE Gauss DE	Gauss/Joint DE Gauss Naïve DE
Inputs → Regressor → Predict real no.			

BCs that have both real and categorical inputs?

	Categorical inputs only	Real-valued inputs only	Mixed Real / Cat okay
Inputs → Classifier → Predict category	Joint BC Naïve BC	Gauss BC	Dec Tree Gauss/Joint BC Gauss Naïve BC
Inputs → Density Estimator → Prob-ability	Joint DE Naïve DE	Gauss DE Gauss DE	Gauss/Joint DE Gauss Naïve DE
Inputs → Regressor → Predict real no.			

Copyright © 2001, Andrew W. Moore

Mixed Categorical / Real Density Estimation

- Write $\mathbf{x} = (\mathbf{u},\mathbf{v}) = (\underbrace{u_1 , u_2 , \ldots u_q}_{\text{Real valued}} , \underbrace{v_1 , v_2 \ldots v_{m-q}}_{\text{Categorical valued}})$

$$P(\mathbf{x} \mid M) = P(\mathbf{u},\mathbf{v} \mid M)$$

(where M is any Density Estimation Model)

Not sure which tasty DE to enjoy? Try our…

Joint / Gauss DE Combo

$$P(\mathbf{u},\mathbf{v}\,|M) = \underbrace{P(\mathbf{u}\,|\mathbf{v}\,,M)}_{\text{Gaussian with parameters depending on } \mathbf{v}}\ \underbrace{P(\mathbf{v}\,|M)}_{\text{Big “m-q”-dimensional lookup table}}$$

MLE learning of the Joint / Gauss DE Combo

$$P(\mathbf{u},\mathbf{v} \mid M) = P(\mathbf{u} \mid \mathbf{v}, M)\, P(\mathbf{v} \mid M)$$

μ_v	=	Mean of **u** among records matching **v**
Σ_v	=	Cov. of **u** among records matching **v**
q_v	=	Fraction of records that match **v**

$$\mathbf{u} \mid \mathbf{v}, M \sim N(\mu_v, \Sigma_v) \quad , P(\mathbf{v} \mid M) = q_v$$

MLE learning of the Joint / Gauss DE Combo

$$P(\mathbf{u},\mathbf{v}\,|M) = P(\mathbf{u}\,|\mathbf{v}\,,M)\;P(\mathbf{v}\,|M)$$

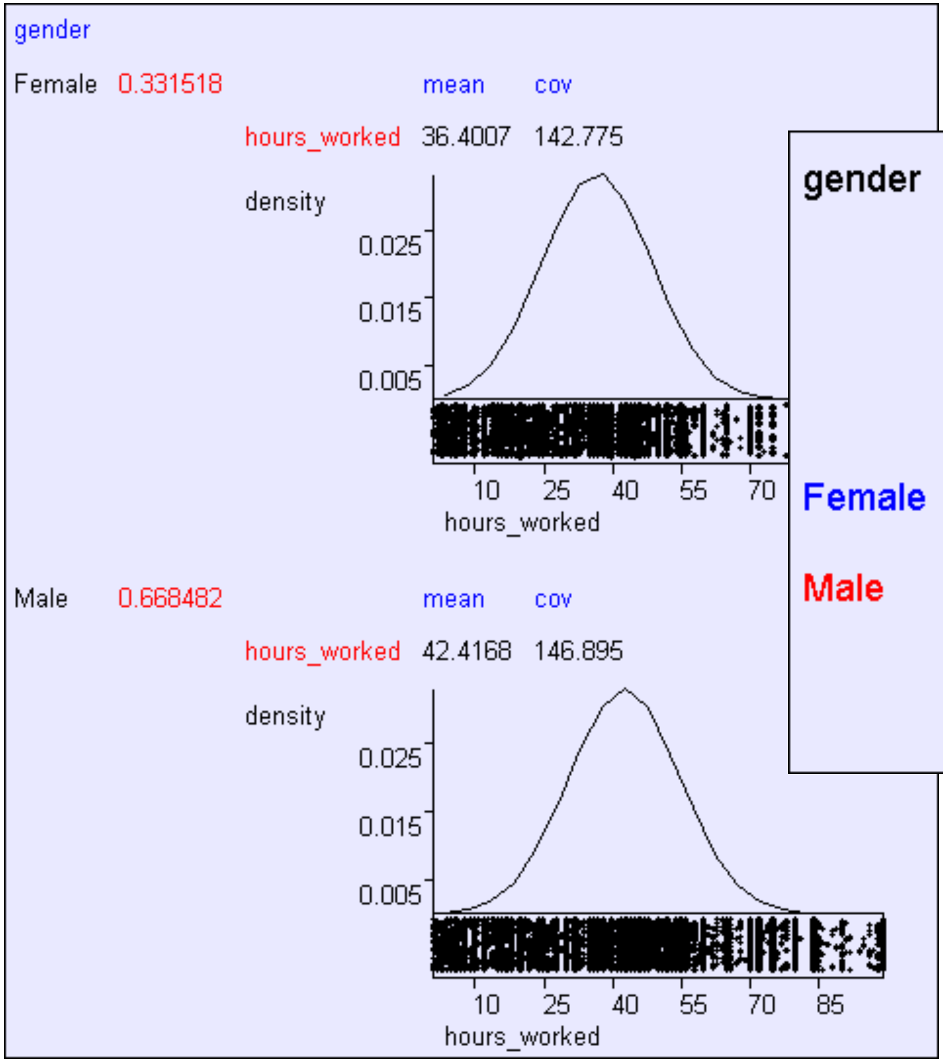
μ_v	=	Mean of **u** among records matching **v**	=	$\dfrac{1}{R_{\mathbf{v}}}\sum_{k \text{ s.t. } \mathbf{v}_k=\mathbf{v}} \mathbf{u}_k$
Σ_v	=	Cov. of **u** among records matching **v**	=	$\dfrac{1}{R_{\mathbf{v}}}\sum_{k \text{ s.t. } \mathbf{v}_k=\mathbf{v}} (\mathbf{u}_k-\boldsymbol{\mu}_{\mathbf{v}})(\mathbf{u}_k-\boldsymbol{\mu}_{\mathbf{v}})^T$
q_v	=	Fraction of records that match **v**	=	$\dfrac{R_{\mathbf{v}}}{R}$

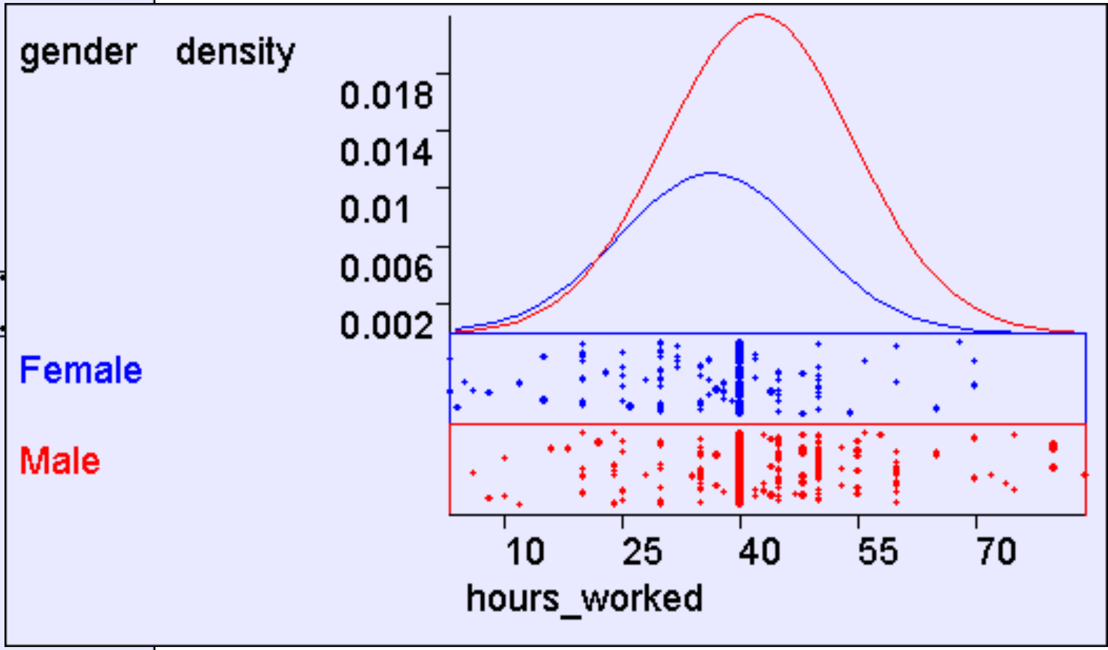
$R_{\mathbf{v}}$ = # records that match **v**

$$\mathbf{u}\,|\mathbf{v}\,,M \sim N(\mu_v\,,\,\Sigma_v)\;\;,\; P(\mathbf{v}\,|M) = q_v$$

Copyright © 2001, Andrew W. Moore

Gender and Hours Worked*





*As with all the results from the UCI "adult census" dataset, we can't draw any real-world conclusions since it's such a non-real-world sample

What we just did $\longrightarrow$ Joint / Gauss DE Combo

Copyright © 2001, Andrew W. Moore

What we do next → Joint / Gauss **BC** Combo

Joint / Gauss **BC** Combo

$$P(Y=i\,|\,\mathbf{u},\mathbf{v})=\frac{p(\mathbf{u},\mathbf{v}\,|\,M_i)P(Y=i)}{p(\mathbf{u},\mathbf{v})}$$

$$=\frac{p(\mathbf{u},|\,\mathbf{v},M_i)\,p(\mathbf{v}\,|\,M_i)P(Y=i)}{p(\mathbf{u},\mathbf{v})}$$

$$=\frac{N(\mathbf{u};\boldsymbol{\mu}_{i,\mathbf{v}},\boldsymbol{\Sigma}_{i,\mathbf{v}})\,q_{i,\mathbf{v}}\;p_i}{p(\mathbf{u},\mathbf{v})}$$

Copyright © 2001, Andrew W. Moore

Joint / Gauss **BC** Combo

$$P(Y=i\,|\,\mathbf{u},\mathbf{v}) = \frac{p(\mathbf{u},\mathbf{v}\,|\,M_i)P(Y=i)}{p(\mathbf{u},\mathbf{v})}$$

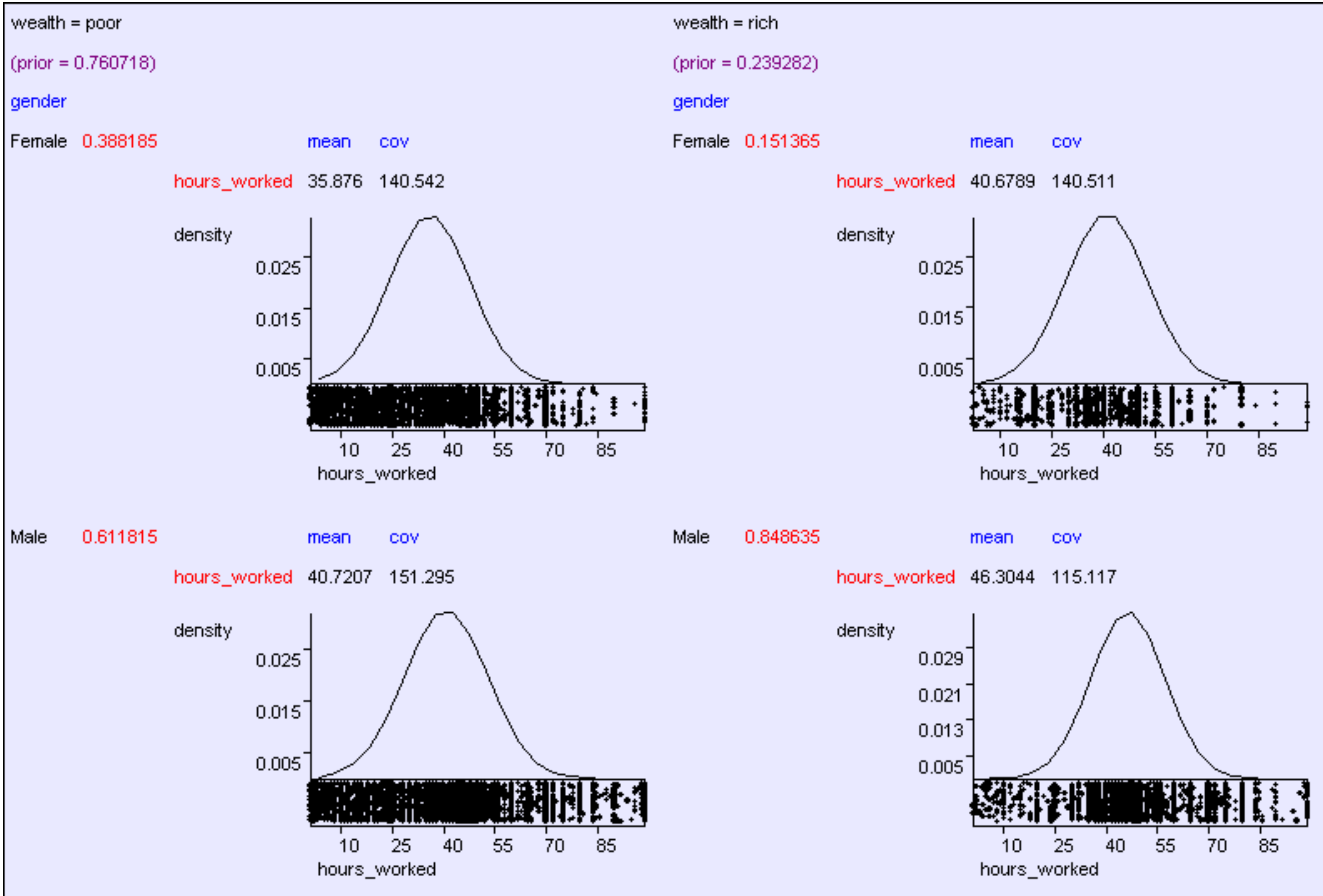
$$= \frac{p(\mathbf{u},|\,\mathbf{v},M_i)\,p(\mathbf{v}\,|\,M_i)P(Y=i)}{p(\mathbf{u},\mathbf{v})}$$

$$= \frac{N(\mathbf{u};\boldsymbol{\mu}_{i,\mathbf{v}},\boldsymbol{\Sigma}_{i,\mathbf{v}})\,q_{i,\mathbf{v}}\;p_i}{p(\mathbf{u},\mathbf{v})}$$

$\mu_{i,v}$	=	Mean of **u** among records matching **v** and in which y=i
$\Sigma_{i,v}$	=	Cov. of **u** among records matching **v** and in which y=i
$q_{i,v}$	=	Fraction of "y=i" records that match **v**
p_i	=	Fraction of records that match "y=i"

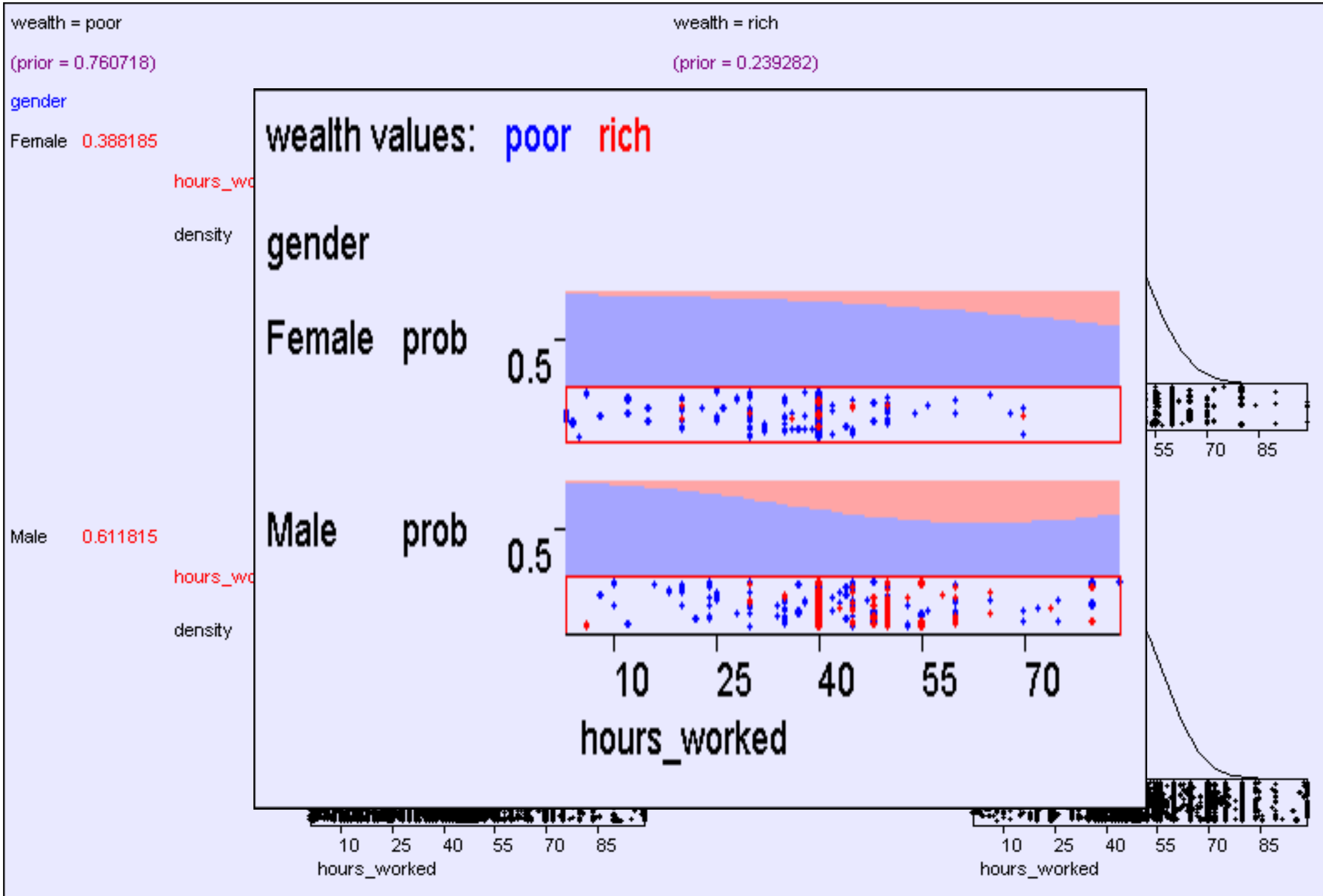
Rather so-so-notation for "Gaussian with mean $\mu_{i,v}$ and covariance $\Sigma_{i,v}$ evaluated at **u**"

Gender, Hours→Wealth



Copyright © 2001, Andrew W. Moore

Gender, Hours→Wealth



Joint / Gauss DE Combo and Joint / Gauss BC Combo: The downside

- (Yawn…we've done this before…)
  More than a few categorical attributes blah blah blah massive table blah blah lots of parameters blah blah just memorize training data blah blah blah do worse on future data blah blah need to be more conservative blah

Naïve/Gauss combo for Density Estimation

Real

Categorical

$$p(\mathbf{u},\mathbf{v}\mid M)=\left(\prod_{j=1}^{q}p(u_j\mid M)\right)\left(\prod_{j=1}^{m-q}P(v_j\mid M)\right)$$

$$u_j \mid M \sim N(\mu_j,\sigma_j^2) \qquad v_j \mid M \sim \text{Multinomial}[q_{j1},q_{j2},\ldots,q_{jN_j}]$$

How many parameters?

Naïve/Gauss combo for Density Estimation

Real Categorical

$$p(\mathbf{u},\mathbf{v} \mid M) = \left(\prod_{j=1}^{q} p(u_j \mid M)\right)\left(\prod_{j=1}^{m-q} P(v_j \mid M)\right)$$

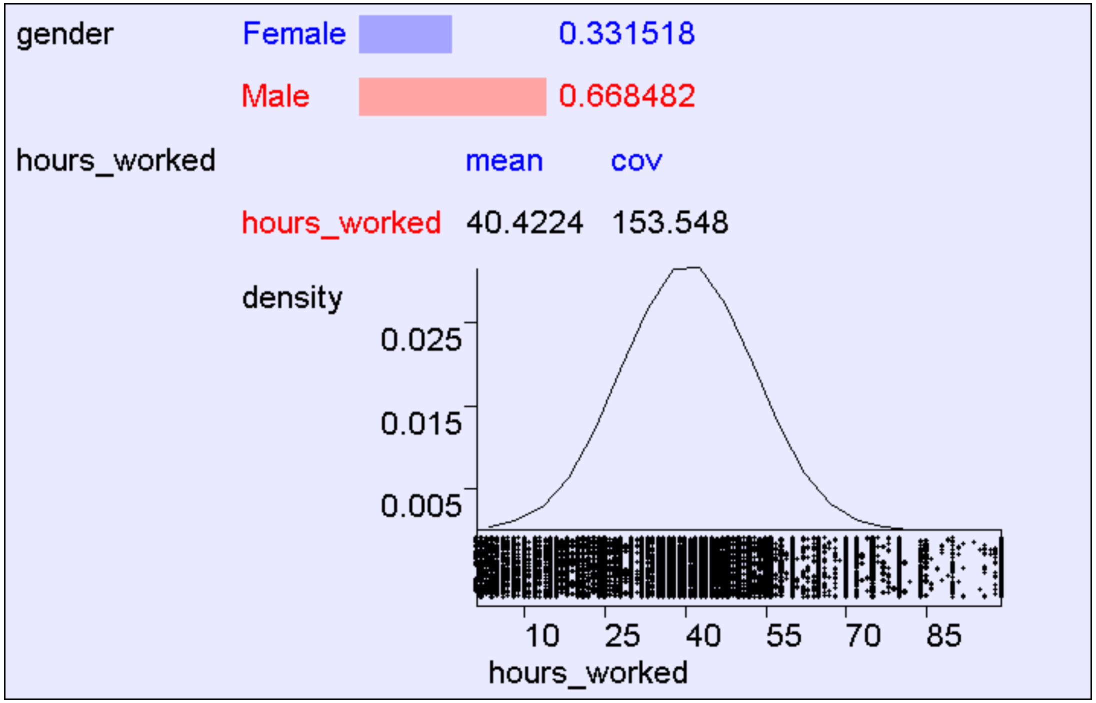
$$u_j \mid M \sim N(\mu_j, \sigma_j^2) \quad v_j \mid M \sim \text{Multinomial}[q_{j1}, q_{j2}, \ldots, q_{jN_j}]$$

$$\mu_j = \frac{1}{R}\sum_k u_{kj}$$

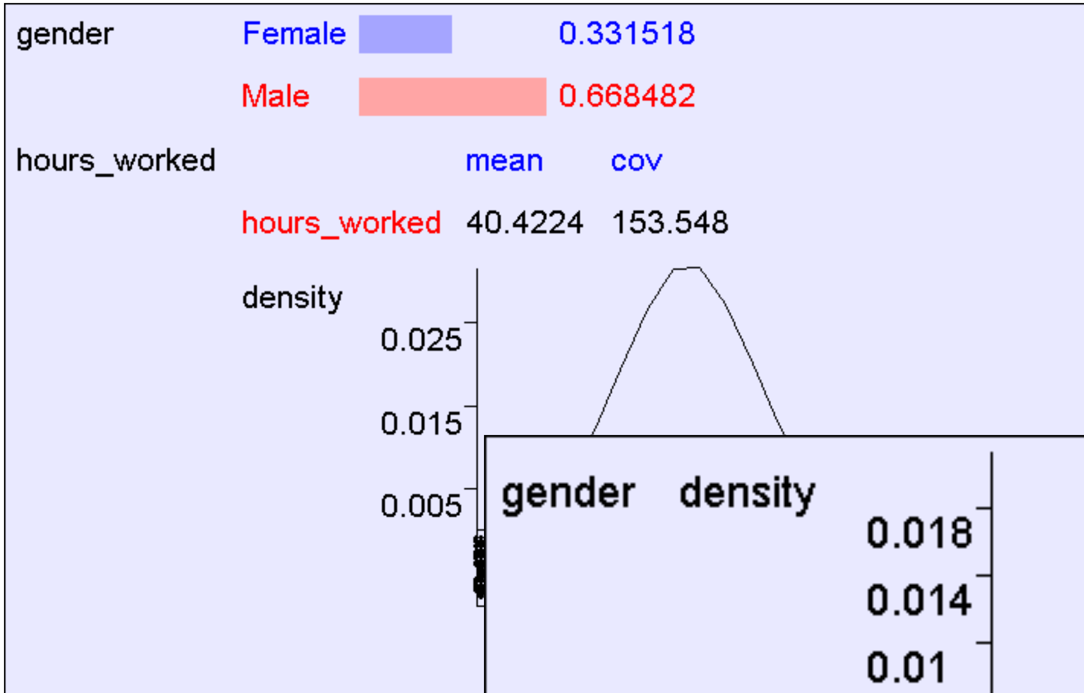
$$\sigma_j^2 = \frac{1}{R}\sum_k (u_{kj} - \mu_j)^2$$

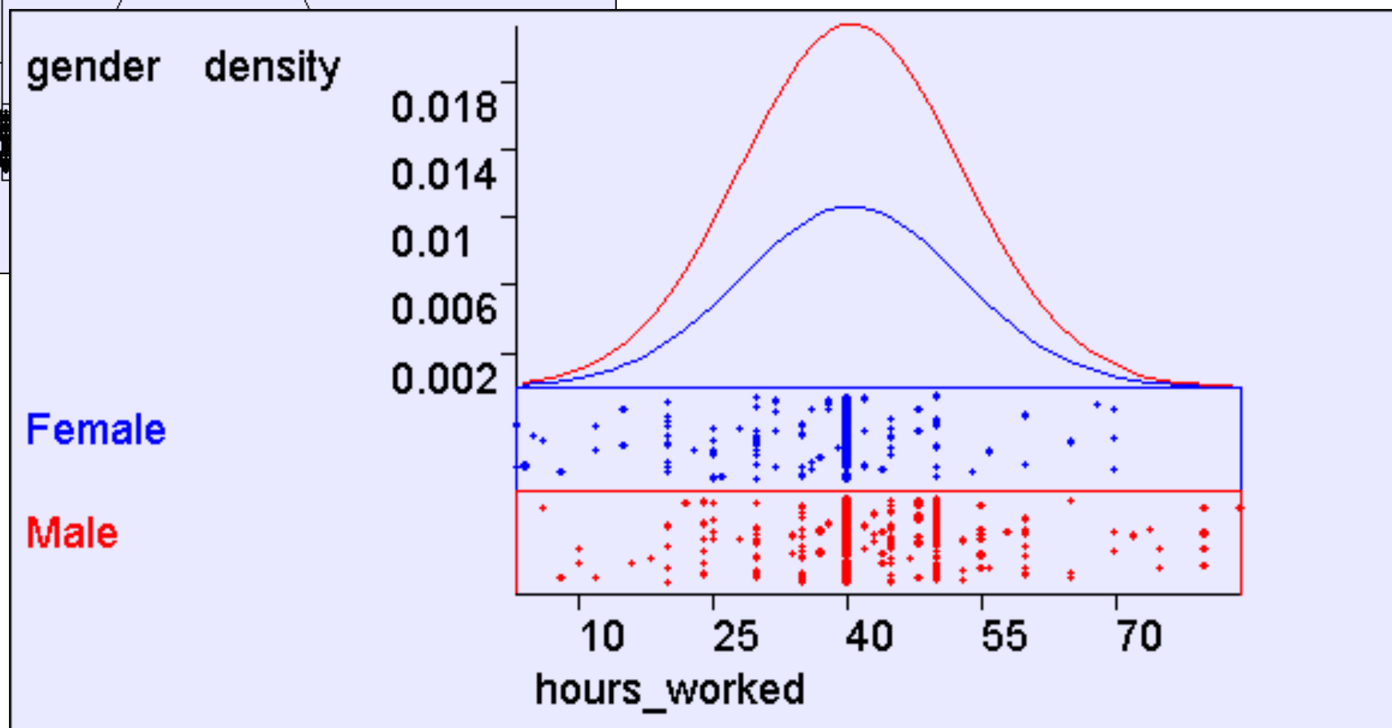
$$q_{jh} = \frac{\#\text{ of records in which } v_j = h}{R}$$

Naïve/Gauss DE Example



Naïve/Gauss DE Example





Naïve / Gauss BC

$$P(Y=i \mid \mathbf{u},\mathbf{v}) = \frac{p(\mathbf{u},\mathbf{v} \mid Y=i)P(Y=i)}{p(\mathbf{u},\mathbf{v})}$$

$$= \frac{1}{p(\mathbf{u},\mathbf{v})} \prod_{j=1}^{q} p(u_j \mid \mu_{ij}, \sigma_{ij}^2) \quad \prod_{j=1}^{m-q} P(v_j \mid \mathbf{q}_{ij}) \quad P(Y=i)$$

$$= \frac{1}{p(\mathbf{u},\mathbf{v})} \prod_{j=1}^{q} N(u_j ; \mu_{ij}, \sigma_{ij}^2) \quad \prod_{j=1}^{m-q} q_{ij}[v_j] \quad p_i$$

μ_{ij}	= Mean of u_j among records in which y=i
σ^2_{ij}	= Var. of u_j among records in which y=i
$q_{ij}[h]$	= Fraction of "y=i" records in which v_j = h
p_i	= Fraction of records that match "y=i"

Gauss / Naïve BC Example

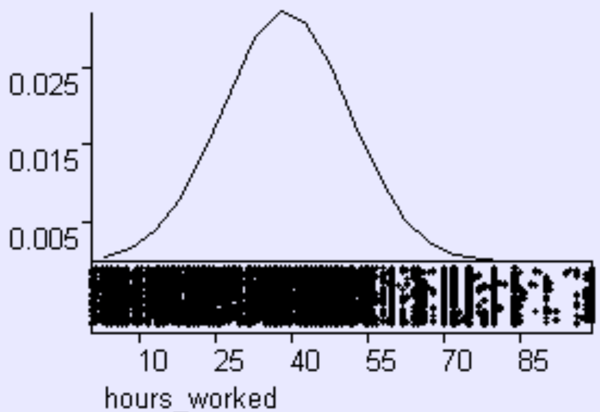
wealth = poor

(prior = 0.760718)

gender		
	Female	0.388185
	Male	0.611815

hours_worked	mean	cov
hours_worked	38.84	152.692

density



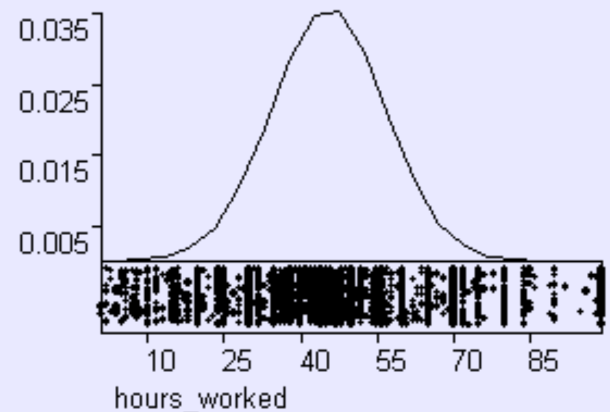
wealth = rich

(prior = 0.239282)

gender		
	Female	0.151365
	Male	0.848635

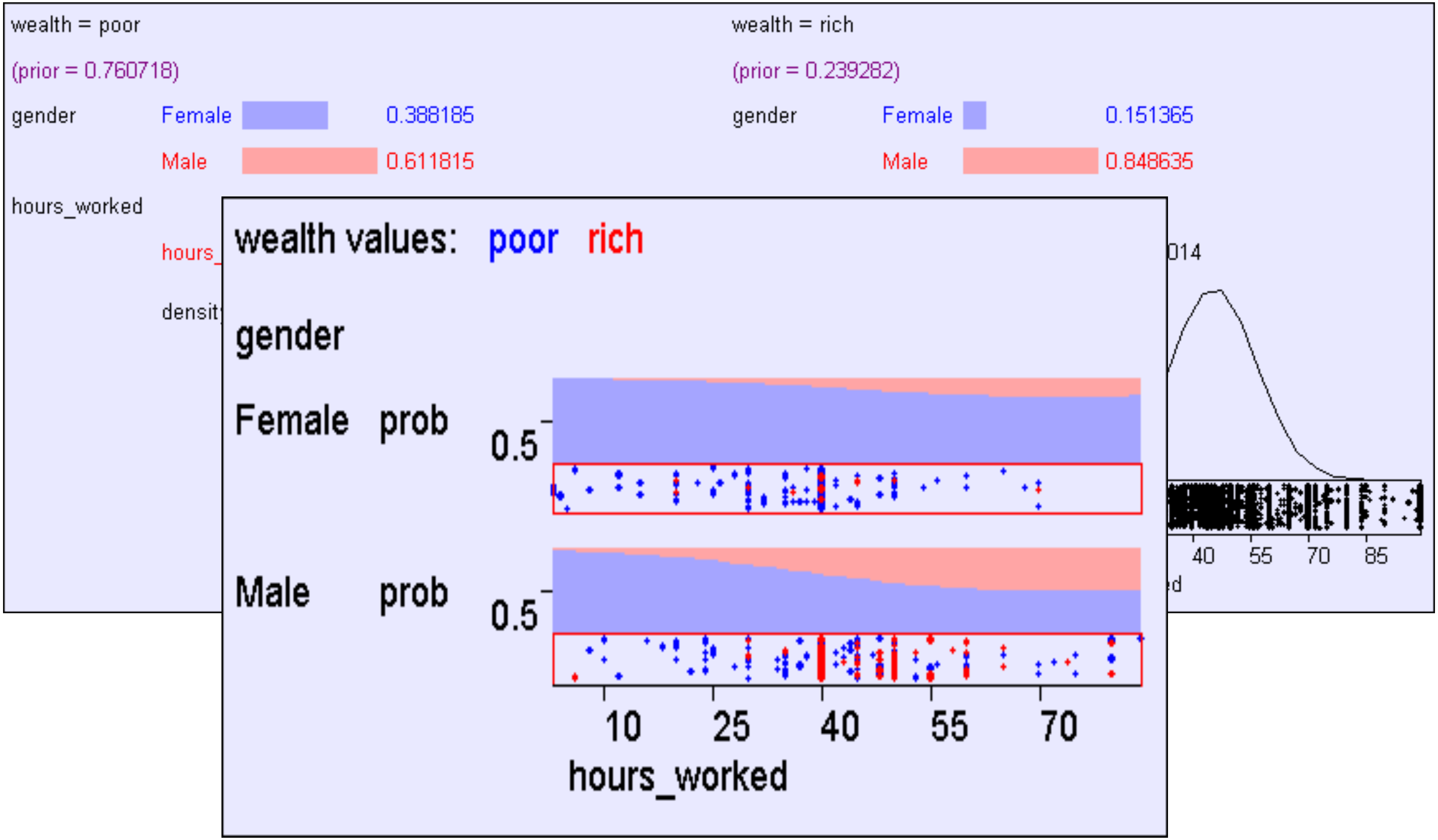
hours_worked	mean	cov
hours_worked	45.4529	123.014

density

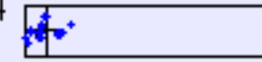
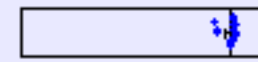
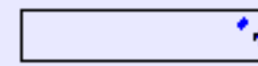


Copyright © 2001, Andrew W. Moore

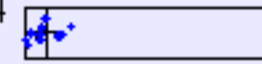
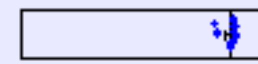
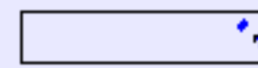
Gauss / Naïve BC Example

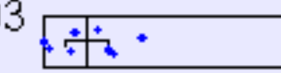
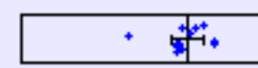
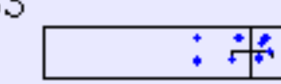


Learn Wealth from 15 attributes

Name	Model	Parameters	FracRight			
Model1	bayesclass	density=joint submodel=gauss gausstype=general	0.718009	+/-	0.00570714	
Model2	bayesclass	density=naive submodel=gauss gausstype=general	0.832234	+/-	0.00288377	
Model3	dtree	max_children=4 ne_splits=y max_pchance=0.05 adjust_chi=y max_nodes=50	0.850702	+/-	0.00364538	

Learn Wealth from 15 attributes

Name	Model	Parameters	FracRight			
Model1	bayesclass	density=joint submodel=gauss gausstype=general	0.718009	+/-	0.00570714	
Model2	bayesclass	density=naive submodel=gauss gausstype=general	0.832234	+/-	0.00288377	
Model3	dtree	max_children=4 ne_splits=y max_pchance=0.05 adjust_chi=y max_nodes=50	0.850702	+/-	0.00364538	

Model	Parameters	FracRight			
bayesclass	density=joint submodel=gauss gausstype=general	0.800418	+/-	0.00321903	
bayesclass	density=naive submodel=gauss gausstype=general	0.819745	+/-	0.00240386	
dtree	max_children=4 ne_splits=y max_pchance=0.05 adjust_chi=y max_nodes=50	0.826113	+/-	0.00327583	

Same data, except all real values discretized to 3 levels

Copyright © 2001, Andrew W. Moore

Learn Race from 15 attributes

Name	Model	Parameters	FracRight
Model1	bayesclass	density=joint submodel=gauss gausstype=general	0.391303 +/- 0.00586792
Model2	bayesclass	density=naive submodel=gauss gausstype=general	0.788686 +/- 0.00560675
Model3	dtree	max_children=4 ne_splits=y max_pchance=0.05 adjust_chi=y max_nodes=50	0.860919 +/- 0.00272011

Copyright © 2001, Andrew W. Moore

What you should know

- A lot of this should have just been a corollary of what you already knew
- Turning Gaussian DEs into Gaussian BCs
- Mixing Categorical and Real-Valued

Copyright © 2001, Andrew W. Moore

Questions to Ponder

- Suppose you wanted to create an example dataset where a BC involving Gaussians crushed decision trees like a bug. What would you do?
- Could you combine Decision Trees and Bayes Classifiers? How? (maybe there is more than one possible way)

Copyright © 2001, Andrew W. Moore