

---

# Introduction to logistic regression

# Outline

---

- General concepts
- Model formulation
- Parameter interpretation
- Estimation
- Model formulation in genetic studies
  - single genotype models
  - multilocus models
  - haplotype models
- A case study in *R*: CEPH data

# General Concepts

---

- Regression is the study of dependence between a response variable ( $y$ ), the dependent variable and one or several predictors ( $x$ ), the independent variables.
- The response variable is binary
- It is a simple representation/modeling of the dependence between one or several variables.

# The Logistic Model

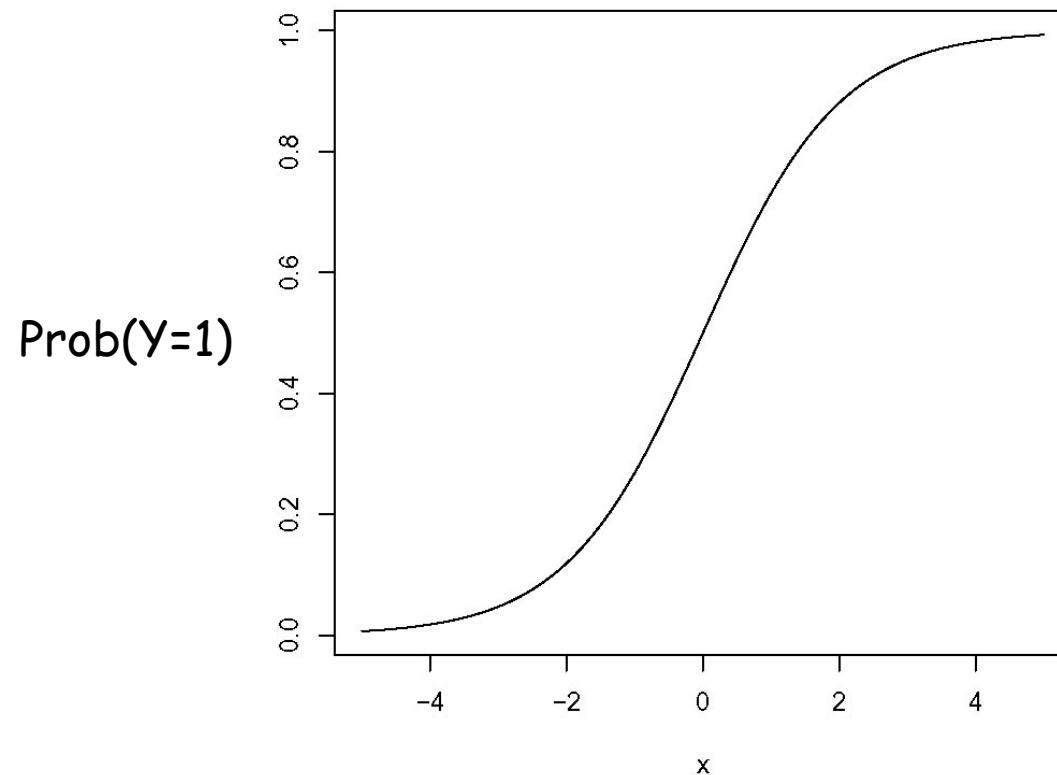
---

- $\text{Prob}(Y_i=1) = \frac{\exp(\eta_i)}{1+\exp(\eta_i)}$   
 $\eta_i = \sum_j x_{ij} b_j$  - Linear Predictor
- $x_{ij}$  - Design Matrix (genotypes etc)
- $b_j$  - Model Parameters (to be estimated)
- Model is investigated by
  - estimating the  $b_j$ 's by maximum likelihood
  - testing if the estimates are different from 0

# The Logistic Function

---

$$\text{Prob}(Y_i=1) = \exp(\eta_i) / (1 + \exp(\eta_i))$$



# Interpretation of the Parameters

- $\text{Logit } \{Y=1|X\} = \text{logit}(P) = \text{logit}[P/(1-P)]$   
 $= X\beta,$

## Measure of risk: $OR = \exp(\beta)$

- Increase  $X_j$  by  $d \rightarrow$  increase odds  $Y=1$  by  $\exp(\beta_j d)$ ,  
increase log odds by  $\beta_j d$
- If there is only one predictor  $X$  and that predictor is binary, the model can be written

$$\text{Logit } \{Y=1|X=0\} = \beta_0$$

$$\text{Logit } \{Y=1|X=1\} = \beta_0 + \beta_1.$$

# Interpretation

-cont'd

- One continuous predictor:

$$\text{Logit } \{Y=1|X\} = \beta_0 + \beta_1 X$$

- Two treatments (indicated by  $X_1=0$  or  $1$ ) and one continuous covariable ( $X_2$ )

$$\text{Logit } \{Y=1|X\} = \beta_0 + \beta_1 X_1 + \beta_2 X_2$$

Eg.,

$$\text{Logit } \{Y=1|X_1=0, X_2\} = \beta_0 + \beta_2 X_2$$

$$\text{Logit } \{Y=1|X_1=1, X_2\} = \beta_0 + \beta_1 + \beta_2 X_2$$

# Estimation

---

- In ordinary least squares regression, main objective function is SSE
- If residuals are normally distributed, the resulting least squares estimates are optimal (consistency and lowest variances among unbiased estimates)
- Other fitting criteria such as minimizing sum of absolute errors are needed for non-normal residuals (or residuals not assumed to be symmetrically distributed)
- **With binary  $Y$  a drastic change is needed**
- Need a general way to write down a good fitting criterion for many different types of  $Y$  and for any distribution of  $Y \mid X$
- **Maximum likelihood (ML) is a general solution**

# Maximum likelihood estimation

---

- **Example: 1-sample binomial problem**
- Single unknown  $P$  = probability of an event in a population  
unknown parameter, the probability of an event in a population.
- Occurrence of the event signaled by  $Y = 1$ , non-occurrence by  $Y = 0$ , for an individual with  $\text{Prob}\{Y = 1\} = P$
- Draw a random sample of size  $n = 3$  from the population and observed the events  $Y = 1; 0; 1$
- Assuming individuals in the sample act completely independently, proba. of the 3 events is  $P^2(1-P)$ ; this joint probability is called the likelihood
- $P$  is unknown but the ML estimate (MLE) can be computed by solving for  $P$  that makes the likelihood maximum

# Maximum likelihood estimation

---

• In general if  $Y$  is binary so that the sample is  $Y_1, \dots, Y_n$  and  $s$  is  $\sum_i Y_i$ , the likelihood is:

$$L = \prod_i P^{Y_i}(1-P)^{1-Y_i} = P^s(1-P)^{n-s}$$

• For numerical and statistical reasons we work with the **log-likelihood** function

$$\log L = s \log(P) + (n - s) \log(1 - P)$$

• In logistic regression we allow differences in subject characteristics through  $X_s$

# Logistic Regression in Genetics

---

- Applicable to Association Studies
- Data:
  - Binary outcomes (eg disease status)
  - Dependent on genotypes [+ sex, environment]
- Aim is to identify which factors influence the outcome
- Rigorous tests of statistical significance
- Flexible modelling language
- Generalisation of Chi-Squared Test

# Example

Research article

Open Access

### SNP-SNP interactions in breast cancer susceptibility

Venüs Ümmiye Onay<sup>1,3</sup>, Laurent Briollais<sup>1,2,5</sup>, Julia A Knight<sup>1,2,5</sup>, Ellen Shi<sup>4</sup>,  
Yuanyuan Wang<sup>1,2</sup>, Sean Wells<sup>1,3</sup>, Hong Li<sup>1,3</sup>, Isaac Rajendram<sup>1,3</sup>,  
Irene L Andrulis<sup>1,3,4,6,7</sup> and Hilmi Ozcelik\*<sup>1,3,7</sup>

Table 4: Analysis of two-way SNP interaction effects on breast cancer.

Two-way Interactions between polymorphisms	Crude P-values†	Bootstrap frequency of stepwise variable selection*	Bootstrap P-values‡	FDR adjusted P-values§	FPRP¶
XPD-[Lys751Gln] and IL10-[G(-1082)A]	0.035	68%	0.001	0.007	0.092
COMT-[Met108/158Val] and CCND1-[Pro241Pro]	0.010	61%	0.002	0.014	0.169
GSTP1-[Ile105Val] and COMT-[Met108/158Val]	0.036	54%	0.001	0.007	0.093
CYP17-[C(518)T] and GADD45-[C(153+168)T]	0.024	53%	0.018	0.062	0.999
BARD1-[Pro24Ser] and ESR1-[Pro325Pro]	0.039	51%	ns‡	ns§	-
TNFA-[G(-308)A] and p27-[Val109Gly]	0.016	49%	0.025	0.079	0.996
BARD1-[Pro24Ser] and p27-[Val109Gly]	0.021	44%	ns‡	ns§	-
BARD1-[Pro24Ser] and XPD-[Lys751Gln]	0.024	36%	0.002	0.014	0.671
ESR1-[Ser10Ser] and ESR1-[Pro325Pro]	0.028	30%	0.097	ns§	0.999

# Coding Unphased Genotypes

---

- Several possibilities:
  - AA, AG, GG original genotypes
  - 12, 21, 22
  - 1, 2, 3
  - 0, 1, 2 # of G alleles
- Missing Data
  - NA default in R

## Case study: genetic association

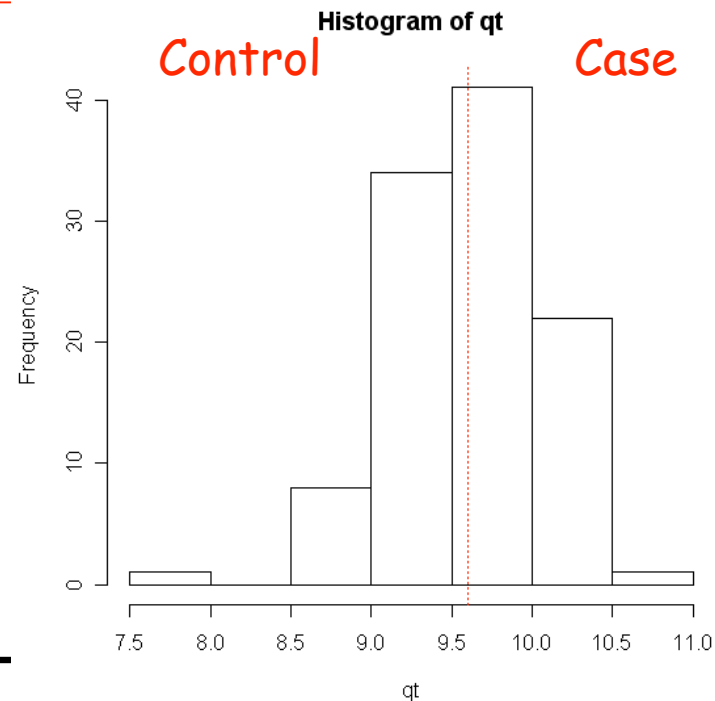
- CEPH pedigrees which consist of 12 multigenerational Caucasian families from Utah including 107 individuals. DNA for the CEPH family pedigree is available for genetic studies.
- The marker data consist of genotypes for 20 SNP markers, six of them are genotyped for all individuals, the remaining 14 are genotyped in only 50-54 individuals.
- The study looks for an association between these SNPs and a gene expression phenotype (mRNA) that we dichotomized here ( $>$ median vs.  $\leq$  median)
- <http://www.sph.umich.edu/csg/abecasis/Merlin/tour/asoc.html>

# Case study: R

## R code:

```
ceph.data<-read.table(paste(my.directory,"ceph_data.txt",sep=""),header=T,na.strings = "0/0")
attach(ceph.data)
##### Create binary trait (Case-Control status) #####
CC<-cut(qt, breaks=c(min(qt),median(qt),max(qt)))
levels(CC)<-c("control","case")
hist(qt)
abline(v=median(qt),col = "red", lty=3)
```

R output



# Case study: genotype association

---

## R code:

```
snp1.geno<-genotype(snp1)
tab.snp1<-table(CC, snp1)
print(tab.snp1)
print(summary(tab.snp1))
```

## R output

	snp1		
CC	1/1	1/3	3/3
control	4	26	23
case	5	24	24

$$OR_{1/1 \text{ vs. } 3/3} = (5 \cdot 23) / (4 \cdot 24) = 1.20$$

$$OR_{1/3 \text{ vs. } 3/3} = (24 \cdot 23) / (26 \cdot 24) = 0.88$$

Number of cases in table: 106

Number of factors: 2

Test for independence of all factors:

Chisq = 0.21239, df = 2, p-value = 0.8993

Chi-squared approximation may be incorrect

# Genotype association: logistic model

## R code:

```
glm.snp1<-glm(CC ~ snp1.geno, family=binomial, data=ceph.data)  
summary(glm.snp1)
```

## R output

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.27352	-1.18256	-0.02969	1.15939	1.21159

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	0.2231	0.6708	0.333	0.739
snp1.geno3/1	-0.3032	0.7281	-0.416	0.677
snp1.geno3/3	-0.1806	0.7315	-0.247	0.805

Problem with the coding!

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 146.95 on 105 degrees of freedom

Residual deviance: 146.73 on 103 degrees of freedom

(1 observation deleted due to missingness)

AIC: 152.73

# Genotype association: logistic model

## Test of association:

Null deviance: 146.95 on 105 degrees of freedom

Residual deviance: 146.73 on 103 degrees of freedom

Likelihood ratio test (LRT) =

$$-2 * (\text{Null deviance} - \text{Residual deviance}) = 0.44$$

To compare with a  $\chi^2$  with 2 d.f.

# Case study: simple genotype association with logistic model

## R code:

```
snp1.geno<-allele.count(snp1.geno,summary(snp1.geno[CC=="control"])[[1]][2])
snp1.geno<-factor(snp1.geno)
glm.snp1<-glm(CC ~ snp1.geno, family=binomial, data=ceph.data)
summary(glm.snp1)
```

## R output

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.27352	-1.18256	-0.02969	1.15939	1.21159

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	0.04256	0.29180	0.146	0.884	OR <sub>1/1 vs. 3/3</sub> = exp(0.18)=1.20
snp1.geno1	-0.12260	0.40654	-0.302	0.763	
snp1.geno2	0.18058	0.73153	0.247	0.805	OR <sub>1/3 vs. 3/3</sub> = exp(-0.12)=0.88

(Dispersion parameter for binomial family taken to be 1)

# Types of genetic effect at a single locus

A = minor allele frequency

	AA	AG	GG
Recessive	0	0	1
Dominant	0	1	1
Additive	0	1	2
Codominant			
var1	0	1	0
var2	0	0	1

# Additive Genotype Model

---

- Code genotypes for SNP1 as
  - 3/3       $x=0$        $\longrightarrow$       **baseline**
  - 3/1       $x=1$
  - 1/1       $x=2$
- Linear Predictor
  - $\eta = b_0 + b_1x$
- $P(Y=1|x) = \exp(b_0 + xb_1)/(1+\exp(b_0 + xb_1))$
- $P_{3/3} = P(Y=1|x=0) = \exp(b_0)/(1+\exp(b_0))$
- $P_{3/1} = P(Y=1|x=1) = \exp(b_0 + b_1)/(1+\exp(b_0 + b_1))$
- $P_{1/1} = P(Y=1|x=2) = \exp(b_0 + 2b_1)/(1+\exp(b_0 + 2b_1))$

# Case study: Additive Genotype Model

R code:

```
snp1.geno<-genotype(snp1)
snp1.geno<-allele.count(snp1.geno,summary(snp1.geno[CC=="control"])[[1]][2])
glm.snp1<-glm(CC ~ snp1.geno, family=binomial, data=ceph.data)
summary(glm.snp1)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-1.177	-1.177	0.000	1.177	1.177

R output

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )	
(Intercept)	-6.924e-17	2.767e-01	-2.50e-16		1
snp1.geno	1.509e-17	3.072e-01	4.91e-17		1

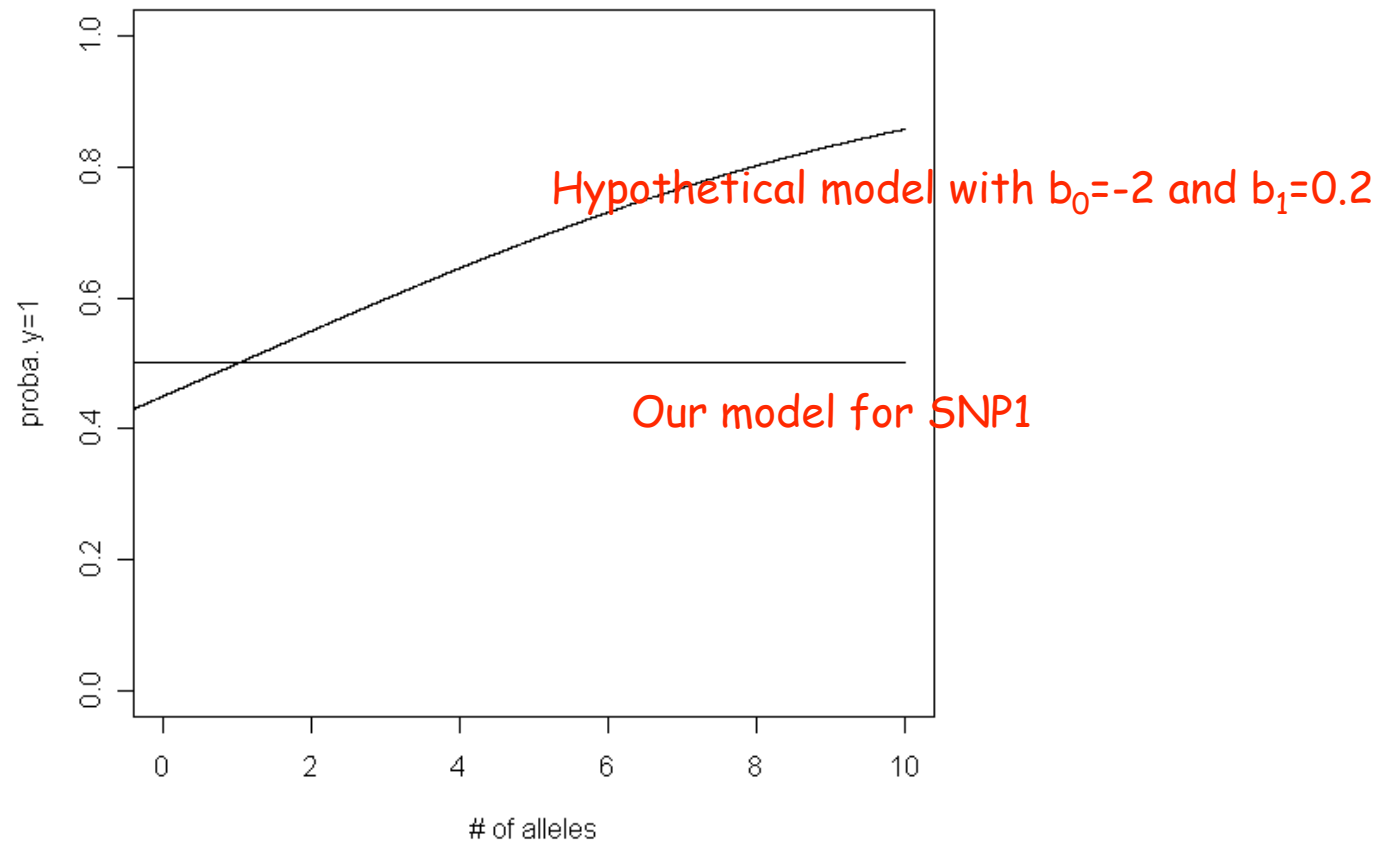
Effect of SNP1 almost null!

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 146.95 on 105 degrees of freedom  
Residual deviance: 146.95 on 104 degrees of freedom  
(1 observation deleted due to missingness)  
AIC: 150.95

# Case study: Additive Genotype Model

---



# Dominant Genotype Model

- Code genotypes for SNP1 as
  - 3/3       $x=0$  → baseline
  - 1/3       $x=1$
  - 1/1       $x=1$
- Linear Predictor
  - $\eta = b_0 + xb_1$
- $P(Y=1|x) = \exp(b_0 + xb_1)/(1+\exp(b_0 + xb_1))$
- $P_{3/3} = P(Y=0|x=0) = \exp(b_0)/(1+\exp(b_0))$
- $P_{1/3} = P_{1/1} = P(Y=0|x=1) = \exp(b_0 + b_1)/(1+\exp(b_0 + b_1))$

# Codominant Genotype Model

---

- Each genotype has a different probability
- Code genotypes as (for example)
  - 3/3       $x_1=0, x_2=0$        $\longrightarrow$  baseline risk
  - 1/3       $x_1=1, x_2=0$
  - 1/1       $x_1=0, x_2=1$
- Linear Predictor
  - $\eta = b_0 + b_1x_1 + b_2x_2$        $\longrightarrow$  two parameters
- $P(Y=1|x) = \frac{\exp(b_0 + xb_1 + yb_2)}{1 + \exp(b_0 + xb_1 + yb_2)}$
- $P_{3/3} = P(Y=1|x_1=0, x_2=0) = \frac{\exp(b_0)}{1 + \exp(b_0)}$
- $P_{1/3} = P(Y=1|x_1=1, x_2=0) = \frac{\exp(b_0 + b_1)}{1 + \exp(b_0 + b_1)}$
- $P_{1/1} = P(Y=1|x_1=0, x_2=1) = \frac{\exp(b_0 + b_2)}{1 + \exp(b_0 + b_2)}$

# Models in R

---

	AA	AG	GG	model	DF
Recessive	0	0	1	$y \sim \text{dominant}(g)$	1
Dominant	0	1	1	$y \sim \text{recessive}(g)$	1
Additive	0	1	2	$y \sim \text{additive}(g)$	1
Genotype				$y \sim$	2
var1	0	1	0	$\text{codominant}(g)$	
var2	0	0	1		

## R code:

```
codominant<-function(x){factor(allele.count(x,summary(x[CC=="control"])[[1]][2]))}
additive<-function(x){allele.count(x,summary(x[CC=="control"])[[1]][2])}
dominant<-function(x){carrier(x,summary(x[CC=="control"])[[1]][2])}
recessive<-function(x){homozygote(x,summary(x[CC=="control"])[[1]][2])}
```

# Data Transformation

---

- `g <- snp marker`
- use these functions to treat a genotype vector in a certain way:
  - `a <- additive(g)`
  - `r <- recessive(g)`
  - `d <- dominant(g)`
  - `c <- codominant(g)`

# Fitting the Model

---

## R code:

- `glm.snp1.codom<-glm(CC ~ snp1.geno.codom, family=binomial, data=ceph.data)`
- `glm.snp1.add<-glm(CC ~ snp1.geno.add, family=binomial, data=ceph.data)`
- `glm.snp1.dom<-glm(CC ~ snp1.geno.dom, family=binomial, data=ceph.data)`
- `glm.snp1.rec<-glm(CC ~ snp1.geno.rec, family=binomial, data=ceph.data)`

- Equivalent models:
  - codominant = dominant + recessive
  - codominant = additive + recessive
  - codominant = additive + dominant

# Model Comparison

---

Akaike criteria (AIC):  $-2 \cdot \ln(L) + 2 \cdot \text{number of parameters}$

Model for SNP1	Df	AIC
Codominant	2	152.73
Additive	1	150.95
Dominant	1	150.91
Recessive	1	150.83

# Scanning all Markers

---

*R* code:

```
for (i in 5:24){
  snp.geno<-codominant(genotype(ceph.data[,i]))
  model.null<-glm(CC ~ 1, family=binomial, data=ceph.data, na.action=na.omit,
subset=!is.na(snp.geno))
  model.alt<-glm(CC ~ snp.geno, family=binomial, data=ceph.data, na.action=na.omit)
  model.anova<-anova(model.null, model.alt)
  snp.stat<-round(model.anova[2,4],3)
  snp.test<-1-pchisq(snp.stat,model.anova[2,3])
  print(c(colnames(ceph.data[i]),round(model.alt$coef,2), round(snp.stat,2),
round(snp.test,2)))
}
```

# Scanning all Markers

---

(Intercept)	snp.geno1	snp.geno2	stat	pval	R output
"snp1"	"0.04"	"-0.12"	"0.18"	"0.21"	"0.9"
"snp2"	"0.86"	"-0.58"	"-0.86"	"1.37"	"0.5"
"snp3"	"0.66"	"-0.47"	"-0.66"	"0.61"	"0.74"
"snp4"	"-0.22"	"0.92"	"15.79"	"5.11"	"0.08"
"snp5"	"0.78"	"-0.67"	"-"	"1.33"	"0.25"
"snp6"	"0.69"	"-0.37"	"-0.29"	"0.4"	"0.82"
"snp7"	"-0.07"	"0.32"	"-0.49"	"2.56"	"0.28"
"snp8"	"0.47"	"0.32"	"-0.69"	"1.75"	"0.42"
"snp9"	"0.75"	"-0.55"	"-0.05"	"0.86"	"0.65"
"snp10"	"0.43"	"-0.09"	"-"	"0.02"	"0.89"
"snp11"	"1.79"	"-2.08"	"-19.36"	"18.6"	"<0.01"
"snp12"	"1.39"	"-0.64"	"-2.37"	"8.12"	"0.02"
"snp13"	"1.39"	"-0.64"	"-2.37"	"8.12"	"0.02"
"snp14"	"0.46"	"0.53"	"0.95"	"2.87"	"0.24"
"snp15"	"1.47"	"-1.18"	"-1.47"	"3.64"	"0.16"
"snp16"	"1.47"	"-1.18"	"-1.47"	"3.64"	"0.16"
"snp17"	"0.14"	"-0.19"	"-0.84"	"1.33"	"0.51"
"snp18"	"1.32"	"-1.46"	"0.47"	"7.22"	"0.03"
"snp19"	"0.6"	"-0.11"	"-0.6"	"0.18"	"0.91"
"snp20"	"0.31"	"-0.28"	"-0.85"	"1.97"	"0.37"

---

# Multilocus Models

---

- Can test the effects of fitting two or more markers simultaneously
- Several multilocus models are possible
- Interaction Model assumes that each combination of genotypes has a different effect
- $CC \sim \text{SNP1} + \text{SNP2} + \text{SNP1} * \text{SNP2} + \dots + \text{SNP}_i + \dots$

# Two-Locus Model with interaction

- $CC \sim \text{SNP1} + \text{SNP2} + \text{SNP1} * \text{SNP2}$

## R code:

```
snp.geno1<-codominant(genotype(snp1))
> snp.geno2<-codominant(genotype(snp2))
> null.model<-glm(CC~ snp.geno1+snp.geno2,family=binomial, data=ceph.data)
> alt.model<-glm(CC~ snp.geno1+snp.geno2+ snp.geno1:snp.geno2,family=binomial, data=ceph.data)
> print(model.anova<-anova(model.null, model.alt))
Analysis of Deviance Table

Model 1: CC ~ 1
Model 2: CC ~ snp.geno
  Resid. Df Resid. Dev  Df Deviance
1     105   146.947
2     103   144.982  2    1.966
> print(snp.test<-1-pchisq(snp.stat,model.anova[2,3]))
[1] 0.3741869
```

# Scanning all two-way interactions

---

## R code:

```
# scanning all two-way interactions

write(c("snp1","snp2","stat","pval"),paste(my.directory,"scan_interactions.txt",sep=""), ncol=4, sep="\t")

for( i in 5:23 ){
k<-i+1

  for( j in k:24 ){
    col1<-colnames(ceph.data)[i]
    col2<-colnames(ceph.data)[j]
    snp.geno1<-dominant(genotype(ceph.data[,i]))
    snp.geno2<-dominant(genotype(ceph.data[,j]))
    model.null<-glm(CC~ snp.geno1+snp.geno2,family=binomial, data=ceph.data, na.action=na.omit)
    model.alt<-glm(CC~ snp.geno1+snp.geno2+ snp.geno1:snp.geno2,family=binomial, data=ceph.data,
na.action=na.omit)
    model.anova<-anova(model.null, model.alt)
    snp.stat<-round(model.anova[2,4],3)
    snp.test<-1-pchisq(snp.stat,model.anova[2,3])
    write(c(col1,col2,snp.stat,snp.test),paste(my.directory,"scan_interactions.txt",sep=""), ncol=4, append=T,
sep="\t" )
  }
}
```

# Multiple Testing

---

- Take care interpreting significance levels when performing multiple tests
- Linkage disequilibrium can reduce the effective number of independent tests
- Permutation is a safe procedure to determine significance
- Repeat  $j=1..N$  times:
  - Permute disease status  $y$  between individuals
  - Fit all markers
  - Record maximum deviance  $\text{maxdev}[j]$  over all markers
- Permutation p-value for a marker is the proportion of times the permuted maximum deviance across all markers exceeds the observed deviance for the marker

# Variable selection

---

## R code:

```
model.fit<-glm(CC~ 1,family=binomial, data=ceph.data)
fit.step <- stepAIC(model.fit, direction="forward", scope = list(upper = ~snp1+snp4+snp14+snp17+snp20, lower = ~1))
```

## R output

Start: AIC=148.95  
CC ~ 1

	Df	Deviance	AIC
+ snp4	2	141.84	147.84
<none>		146.95	148.95
+ snp14	2	144.08	150.08
+ snp20	2	144.98	150.98
+ snp17	2	145.62	151.62
+ snp1	2	146.74	152.74

Step1: AIC=147.84  
CC ~ snp4

	Df	Deviance	AIC
<none>		141.84	147.84
+ snp17	2	139.64	149.64
+ snp20	2	139.81	149.81
+ snp14	2	140.31	150.31
+ snp1	2	141.72	151.72

# Assessing model fit

---

- Hosmer-Lemeshow test is a commonly used test of goodness-of-fit of a binary logistic model
- Idea: compares proportion of events with mean predicted probability within deciles of predicted P

## Problems:

- Arbitrary (number of groups, how to form groups)
- Low power (too many d.f.)

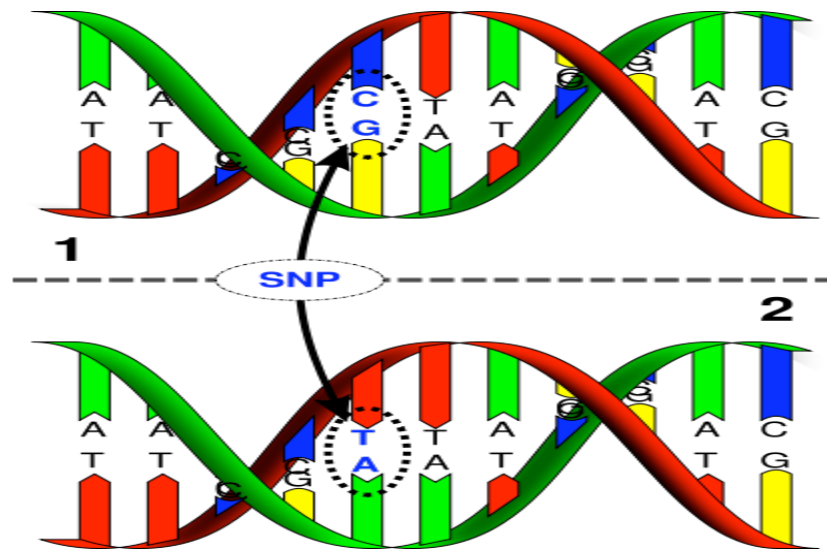
# Model Validation

---

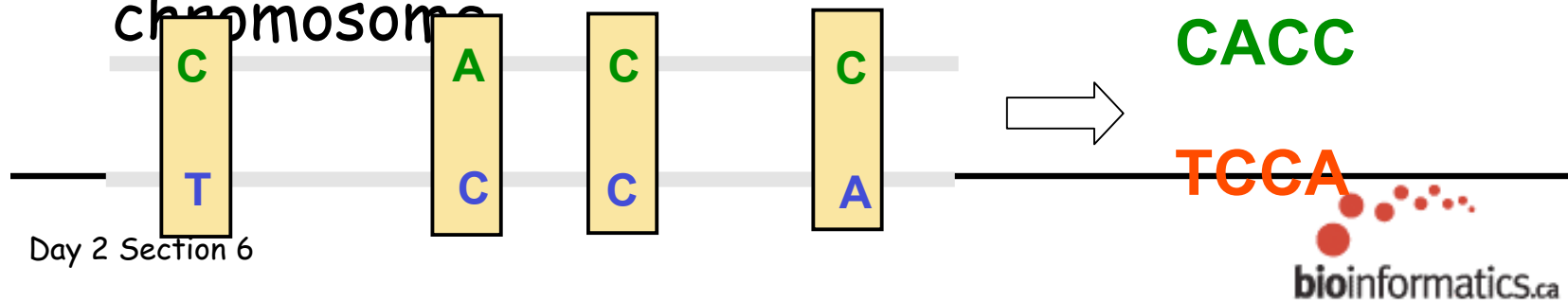
- Assess prediction in a new dataset
- use resampling techniques (Bootstrap, Cross-validation)

# SNP and Haplotype

- DNA strand 1 differs from DNA strand 2 at a single base-pair location (a C/T polymorphism).



- Haplotype: the ordered allele sequence on a chromosome



# Haplotype Association

---

- Haplotype Association
  - Different from multiple genotype models
  - Phase taken into account
  - Haplotype association can be modelled in a similar logistic framework
- Treat haplotypes as extended alleles
- Fit additive, recessive, dominant & genotype models as before
  - Eg haplotypes are  $h = AAGCAT, ATGCTT, \text{etc}$
  - $y \sim \text{additive}(h)$
  - $y \sim \text{dominant}(h) \text{ etc}$

# Haplotype association

---

## R code:

```
##### Haplotype analysis #####

snp1.a1<-allele(genotype(snp1),which=1)
snp1.a2<-allele(genotype(snp1),which=2)
snp2.a1<-allele(genotype(snp2),which=1)
snp2.a2<-allele(genotype(snp2),which=2)

haplo.snp1snp2<-data.frame(snp1.a1,snp1.a2,snp2.a1,snp2.a2)
haplo.mat<-setupGeno(haplo.snp1snp2, miss.val=c(0,NA))
haplo.data<-data.frame(haplo.mat,CC)

minhapfreq<-0.01
rarehap<-F

glm.haplo12<- haplo.glm(CC ~ haplo.mat, na.action="na.geno.keep",allele.lev=attributes(haplo.mat)$unique.alleles,
control=haplo.glm.control(haplo.freq.min=minhapfreq,keep.rare.haplo=rarehap),family = binomial,data=haplo.data)
print(glm.haplo12)
```

# Haplotype analysis

## R output

Call:

```
haplo.glm(formula = CC ~ haplo.mat, family = binomial, data = haplo.data, na.action =  
"na.geno.keep", allele.lev = attributes(haplo.mat)$unique.alleles,  
control = haplo.glm.control(haplo.freq.min = minhapfreq, keep.rare.haplo =  
rarehap))
```

Coefficients:

	coef	se	t.stat	pval
(Intercept)	-7.7e-09	0.277	-2.78e-08	1
haplo.mat	1.2e-08	0.307	3.90e-08	1

Only 1 haplo created !

Haplotypes:

	loc.1	loc.2	hap.freq
haplo.mat.2	1	3	0.321
haplo.base	3	2	0.679

Note: Use weights (haplo freq.) to compute the predicted values

# Summary

---

- Various genetic models possible
- Choice of genotype or haplotype based analyses
- Variable selection can help localizing the disease locus
- Multilocus are difficult to fit: many variables
- Model validation is important