Lecture 9
Gene Prediction

Saleet Jafri

BINF 630

Gene Prediction

- Analysis by sequence similarity can only reliably identify about 30% of the protein-coding genes in a genome
- 50-80% of new genes identified have a partial, marginal, or unidentified homolog
- Frequently expressed genes tend to be more easily identifiable by homology than rarely expressed genes

Gene Finding

- Process of identifying potential coding regions in an uncharacterized region of the genome
- Still a subject of active research
- There are many different gene finding software packages and no one program is capable of finding everything

Genes aren't the only thing we're looking for

•Biologically significant sites include:

•Splice sites

•Protein binding sites (promotors, histones, etc.)

•DNA 3D structure features

•etc.

In a lot of cases, we don't even know what constitutes one of these sites, so all we can do is look for repeating patterns

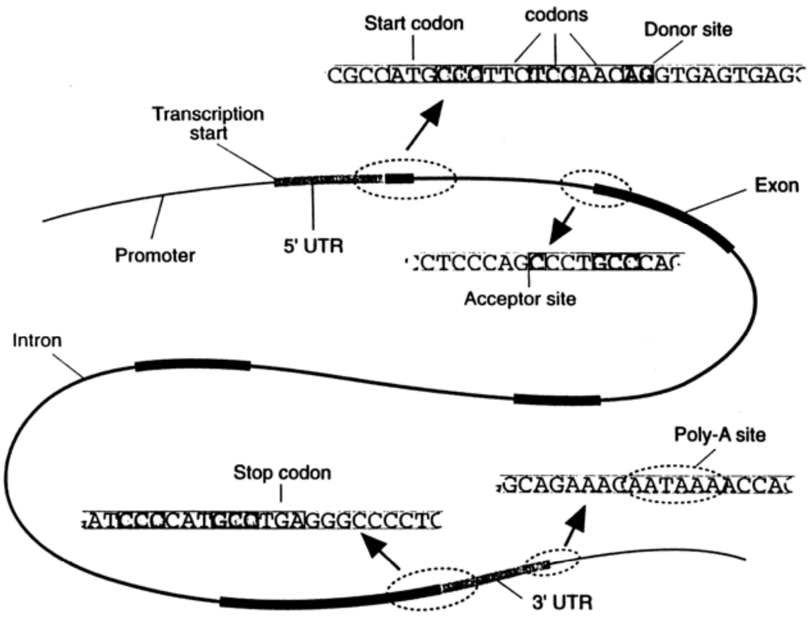


Fig. 8. The structure of a gene with some of the important signals shown.

Eukaryotes vs Prokaryotes

- Eukaryotic DNA wrapped around histones that might result in repeated patterns (periodicity of 10) for histone binding. The promotor regions might be near these sites so that they remain hidden.
- Prokaryotes have no introns.
- Promotor regions and start sites more highly conserved in Prokaryotes
- Different codon use frequencies

Gene finding is species-specific

- Codon usage patterns vary by species
- Functional regions (promoters, splice sites, translation initiation sites, termination signals) vary by species
- Common repeat sequences are species-specific
- Gene finding programs rely on this information to identify coding regions

The genetic code

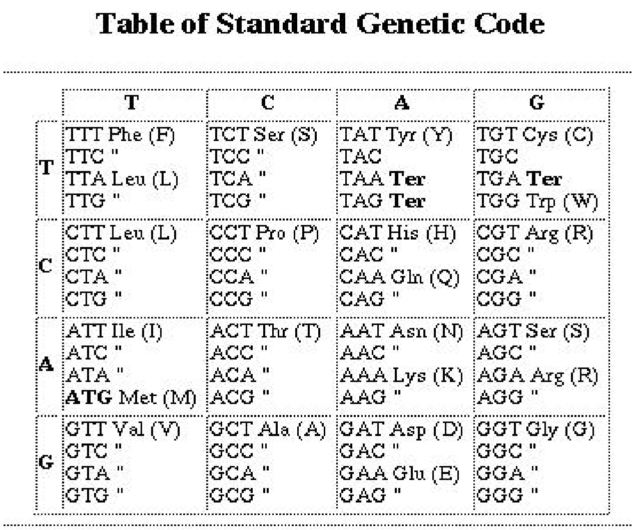
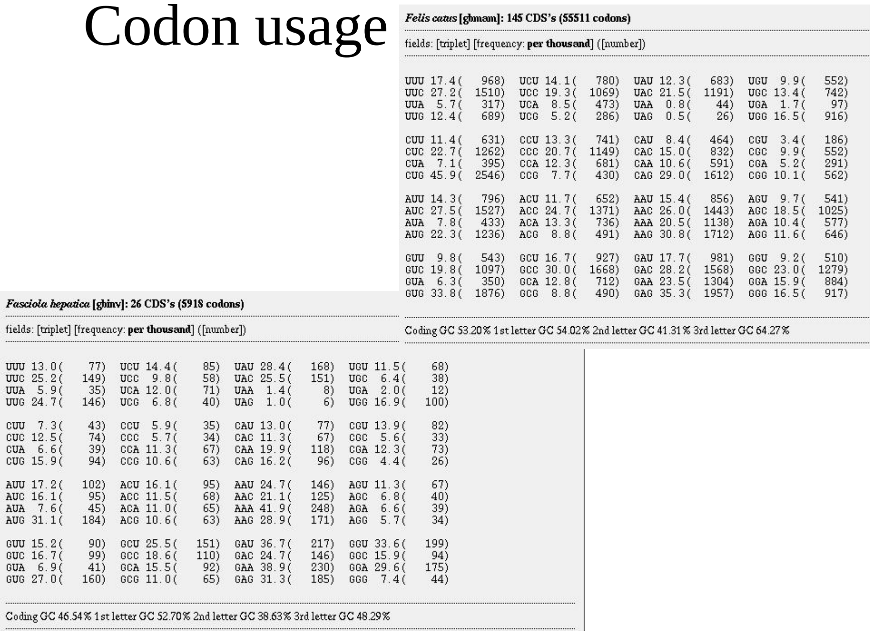


Table of Standard Genetic Code

	T	C	A	G
T	TTT Phe (F) TTC " TTA Leu (L) TTG "	TCT Ser (S) TCC " TCA " TCG "	TAT Tyr (Y) TAC TAA **Ter** TAG **Ter**	TGT Cys (C) TGC TGA **Ter** TGG Trp (W)
C	CTT Leu (L) CTC " CTA " CTG "	CCT Pro (P) CCC " CCA " CCG "	CAT His (H) CAC " CAA Gln (Q) CAG "	CGT Arg (R) CGC " CGA " CGG "
A	ATT Ile (I) ATC " ATA " **ATG** Met (M)	ACT Thr (T) ACC " ACA " ACG "	AAT Asn (N) AAC " AAA Lys (K) AAG "	AGT Ser (S) AGC " AGA Arg (R) AGG "
G	GTT Val (V) GTC " GTA " GTG "	GCT Ala (A) GCC " GCA " GCG "	GAT Asp (D) GAC " GAA Glu (E) GAG "	GGT Gly (G) GGC " GGA " GGG "

Codon usage



***Felis catus* [gbmam]: 145 CDS's (55511 codons)**

fields: [triplet] [frequency: **per thousand**] ([number])

```
UUU 17.4(   968)  UCU 14.1(   780)  UAU 12.3(   683)  UGU  9.9(   552)
UUC 27.2(  1510)  UCC 19.3(  1069)  UAC 21.5(  1191)  UGC 13.4(   742)
UUA  5.7(   317)  UCA  8.5(   473)  UAA  0.8(    44)  UGA  1.7(    97)
UUG 12.4(   689)  UCG  5.2(   286)  UAG  0.5(    26)  UGG 16.5(   916)

CUU 11.4(   631)  CCU 13.3(   741)  CAU  8.4(   464)  CGU  3.4(   186)
CUC 22.7(  1262)  CCC 20.7(  1149)  CAC 15.0(   832)  CGC  9.9(   552)
CUA  7.1(   395)  CCA 12.3(   681)  CAA 10.6(   591)  CGA  5.2(   291)
CUG 45.9(  2546)  CCG  7.7(   430)  CAG 29.0(  1612)  CGG 10.1(   562)

AUU 14.3(   796)  ACU 11.7(   652)  AAU 15.4(   856)  AGU  9.7(   541)
AUC 27.5(  1527)  ACC 24.7(  1371)  AAC 26.0(  1443)  AGC 18.5(  1025)
AUA  7.8(   433)  ACA 13.3(   736)  AAA 20.5(  1138)  AGA 10.4(   577)
AUG 22.3(  1236)  ACG  8.8(   491)  AAG 30.8(  1712)  AGG 11.6(   646)

GUU  9.8(   543)  GCU 16.7(   927)  GAU 17.7(   981)  GGU  9.2(   510)
GUC 19.8(  1097)  GCC 30.0(  1668)  GAC 28.2(  1568)  GGC 23.0(  1279)
GUA  6.3(   350)  GCA 12.8(   712)  GAA 23.5(  1304)  GGA 15.9(   884)
GUG 33.8(  1876)  GCG  8.8(   490)  GAG 35.3(  1957)  GGG 16.5(   917)
```

Coding GC 53.20% 1st letter GC 54.02% 2nd letter GC 41.31% 3rd letter GC 64.27%

***Fasciola hepatica* [gbinv]: 26 CDS's (5918 codons)**

fields: [triplet] [frequency: **per thousand**] ([number])

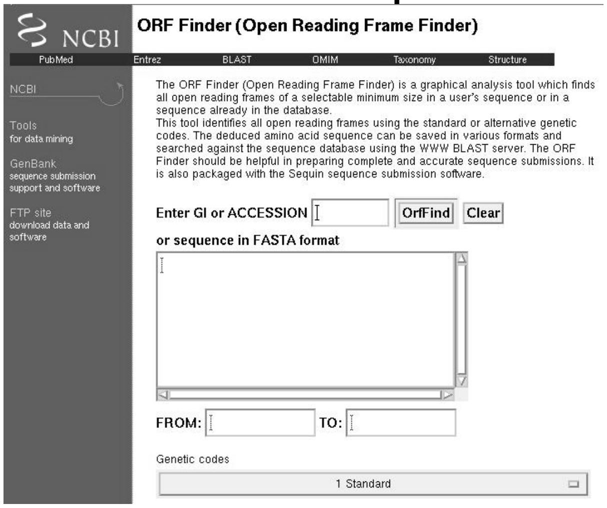
```
UUU 13.0(    77)  UCU 14.4(    85)  UAU 28.4(   168)  UGU 11.5(    68)
UUC 25.2(   149)  UCC  9.8(    58)  UAC 25.5(   151)  UGC  6.4(    38)
UUA  5.9(    35)  UCA 12.0(    71)  UAA  1.4(     8)  UGA  2.0(    12)
UUG 24.7(   146)  UCG  6.8(    40)  UAG  1.0(     6)  UGG 16.9(   100)

CUU  7.3(    43)  CCU  5.9(    35)  CAU 13.0(    77)  CGU 13.9(    82)
CUC 12.5(    74)  CCC  5.7(    34)  CAC 11.3(    67)  CGC  5.6(    33)
CUA  6.6(    39)  CCA 11.3(    67)  CAA 19.9(   118)  CGA 12.3(    73)
CUG 15.9(    94)  CCG 10.6(    63)  CAG 16.2(    96)  CGG  4.4(    26)

AUU 17.2(   102)  ACU 16.1(    95)  AAU 24.7(   146)  AGU 11.3(    67)
AUC 16.1(    95)  ACC 11.5(    68)  AAC 21.1(   125)  AGC  6.8(    40)
AUA  7.6(    45)  ACA 11.0(    65)  AAA 41.9(   248)  AGA  6.6(    39)
AUG 31.1(   184)  ACG 10.6(    63)  AAG 28.9(   171)  AGG  5.7(    34)

GUU 15.2(    90)  GCU 25.5(   151)  GAU 36.7(   217)  GGU 33.6(   199)
GUC 16.7(    99)  GCC 18.6(   110)  GAC 24.7(   146)  GGC 15.9(    94)
GUA  6.9(    41)  GCA 15.5(    92)  GAA 38.9(   230)  GGA 29.6(   175)
GUG 27.0(   160)  GCG 11.0(    65)  GAG 31.3(   185)  GGG  7.4(    44)
```

Coding GC 46.54% 1st letter GC 52.70% 2nd letter GC 38.63% 3rd letter GC 48.29%

Identifying ORFs

- Simple first step in gene finding
- Translate genomic sequence in six frames. Identify stop codons in each frame
- Regions without stop codons are called "open reading frames" or ORFs
- Locate and tag all of the likely ORFs in a sequence
- The longest ORF from a Met codon is a good prediction of a protein encoding sequence.
- SOFTWARE: NCBI ORF Finder

ORF Finder input



ORF Finder (Open Reading Frame Finder)

The ORF Finder (Open Reading Frame Finder) is a graphical analysis tool which finds all open reading frames of a selectable minimum size in a user's sequence or in a sequence already in the database.
This tool identifies all open reading frames using the standard or alternative genetic codes. The deduced amino acid sequence can be saved in various formats and searched against the sequence database using the WWW BLAST server. The ORF Finder should be helpful in preparing complete and accurate sequence submissions. It is also packaged with the Sequin sequence submission software.

Enter GI or ACCESSION

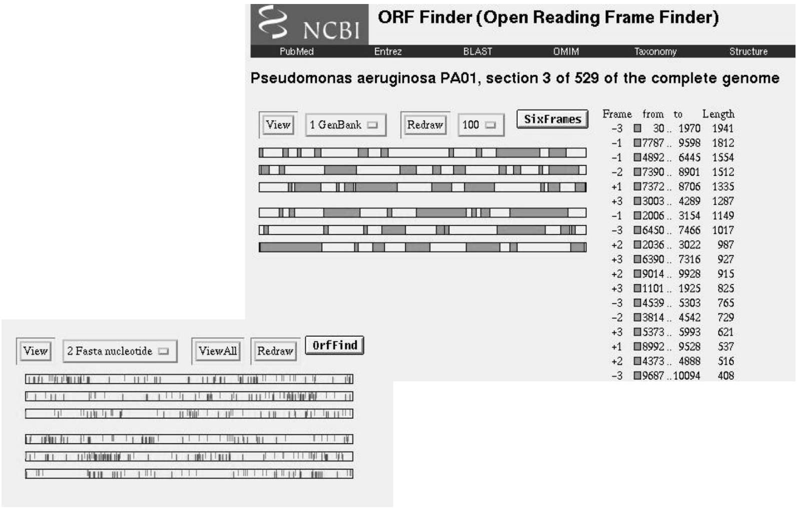
or sequence in FASTA format

FROM: TO:

Genetic codes

1 Standard

ORF finder results



ORF Finder (Open Reading Frame Finder)

Pseudomonas aeruginosa PA01, section 3 of 529 of the complete genome

Frame	from	to	Length
−3	30 ..	1970	1941
−1	7787 ..	9598	1812
−1	4892 ..	6445	1554
−2	7390 ..	8901	1512
+1	7372 ..	8706	1335
+3	3003 ..	4289	1287
−1	2006 ..	3154	1149
−3	6450 ..	7466	1017
+2	2036 ..	3022	987
+3	6390 ..	7316	927
+2	9014 ..	9928	915
+3	1101 ..	1925	825
−3	4539 ..	5303	765
−2	3814 ..	4542	729
+3	5373 ..	5993	621
+1	8992 ..	9528	537
+2	4373 ..	4888	516
−3	9687 ..	10094	408

Tests of the Predicted ORF

- Check if the third base in the codons tends to be the same one more often than by chance alone.
- Are the codons used in the ORF the same as those used in other genes (need codon usage frequency).
- Compare the amino acid sequence for similarity with other know amino acid sequences.

Problems with ORF finding

- A single-character sequencing error can hide a stop codon or insert a false stop codon, preventing accurate identification of ORFs
- Short exons can be overlooked
- Multiple transcripts or ORFs on complementary strand can confuse results

Pattern-based gene finding

- ORF finding based on start and stop codon frequency is a pattern-based procedure
- Other pattern-based procedures recognize characteristic sequences associated with known features and genes, such as ribosome binding sites, promoter sites, histone binding sites, etc.
- Statistically based.

Content-based gene finding

- Content-based gene finding methods rely on statistical information derived from known sequences to predict unknown genes
- Some evaluative measures include: "coding potential" (based on codon bias), periodicity in the sequence, sequence homogeneity, etc.

A standard content-based alignment procedure

- Select a window of DNA sequence from the unknown. The window is usually around 100 base pairs long
- Evaluate the window's potential as a gene, based on a variety of factors
- Move the window over by one base
- Repeat procedure until end of sequence is reached; report continuous high-scoring regions as putative genes

Combining measures

- Programs rarely use one measure to predict genes
- Different values are combined (using probabilistic methods, discriminant analysis, neural net methods, etc.) to produce one "score" for the entire window

Drawbacks to window-based evaluation

- A sequence length of at least 100 b.p. is required before significant information can be gained from the analysis
- Results in a +/- 100 b.p. uncertainty in the start site of predicted coding regions, unless an unambiguous pattern can also be found to indicate the start.

Most are web-based, but...

- Submit sequence; input sequence length may be limited
- Select parameters, if any
- Interpret results
- Most software is first or second generation; results come in non-graphical formats.

GRAIL

- Gene finder for human, mouse, arabidopsis, drosophila, E. coli
- Based on neural networks
- Masks human and mouse repetetive elements
- Incorporates pattern-based searches for several types of promoters and simple repeats
- Accuracy in 75-95% range

Glimmer

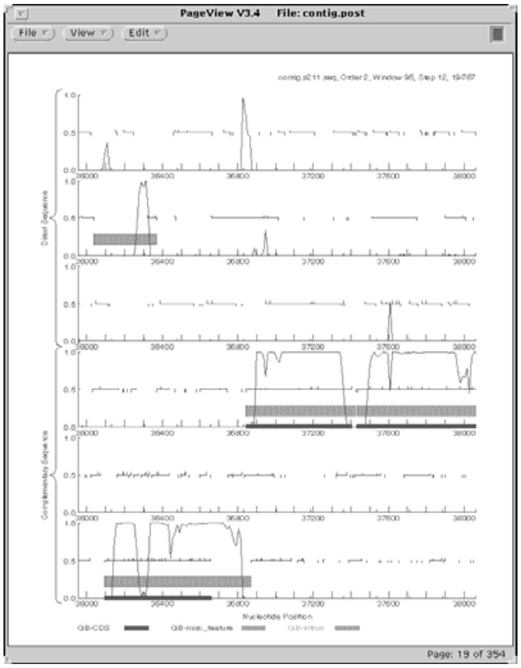
- Genefinder for bacterial and archaebacterial genomes
- Uses an "interpolated Markov model" approach (a Markov model is a model for computing probabilities in the context of sequential events)
- Predicts genes with around 98% accuracy when compared with published annotations
- No web server

GENSCAN

- Genefinder for human and vertebrate sequences
- Probabilistic method based on known genome structure and composition: number of exons per gene, exon size distributions, hexamer composition, etc.
- Only protein coding genes predicted
- Maize and arabidopsis-optimized versions now available
- Accuracy in 50-95% range

GeneMark

- Gene finder for bacterial and archaebacterial sequences
- Markov model-based
- GeneMark and GeneMarkHMM available as web servers
- Accuracy in 90-99% range



CRITICA

- Gene finder for bacterial and archaebacterial genomes
- Combines sequence homology-based prediction with content-based statistical (dicodon probability) analysis
- Accuracy in 90-99% range
- No web server

GeneParser

- Predicts the most likely combination of exons and introns using dynamic programming.
- The intron an exon positions are aligned subject to the constraint that they alternate.
- A neural network is used to adjust the weights given to the sequence indicators of know exon and intron regions such as codon usage, information content, length distribution, hexamer frequencies, and scoring matrices.

Other software

- Generation
- GeneID
- Genie
- GenView
- EcoParse
- etc...

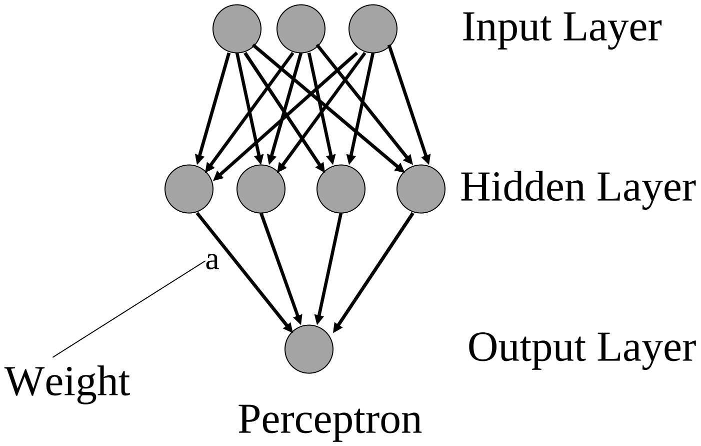
tRNAscan

- Locating tRNA genes is less difficult than other types of gene identification
- pol III promoter is simple; RNA secondary structure is conserved
- SOFTWARE: tRNAscan-SE

Gene finding strategy for beginners

- Choose the appropriate type of gene finder! Make sure that you're using gene finders for microbial (intronless) sequences only to analyze bacteria and archaea!
- If there is no organism-specific gene finder for your system, at least use one that makes sense (i.e. use an arabidopsis gene finder for other plants)

Neural Network Topology



Input Layer

Hidden Layer

Output Layer

a

Weight

Perceptron

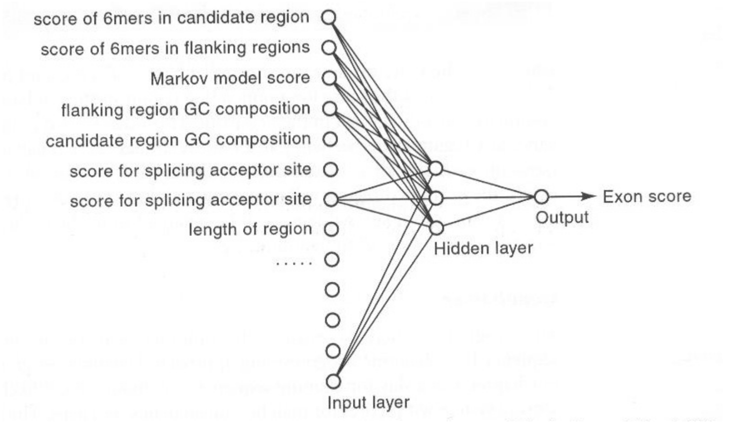
Making Neural Networks

- Take known data and divide into two sets: the *training* set and *test* set.
- Use the optimize the weights so that the neural net gives the best outputs for the training set.
- Test the neural net with the test set to see if it works
- If data is limited, you can permute the data so that you have multiple training and test sets

Caveats with Neural Nets

- The net only performs as well as the training set.
- In other words, it can only find things it is trained to do.
- As more diverse data becomes available, the neural net gets better

Grail II Neural Net


score of 6mers in candidate region
score of 6mers in flanking regions
Markov model score
flanking region GC composition
candidate region GC composition
score for splicing acceptor site
score for splicing acceptor site
length of region
.....
Hidden layer
Output
Exon score
Input layer

- Finds exons in eukaryotic genes, that is, takes inputs and predicts if a gene is present.

Markov Model

- A process is Markov if it has no memory, that is, if the next state it assumes, depends only on its present state and not on any previous states.
- The states can be observed and the transition probabilities between states is known
- Example – rolling a die has 6 possible states each with a probability of 1/6

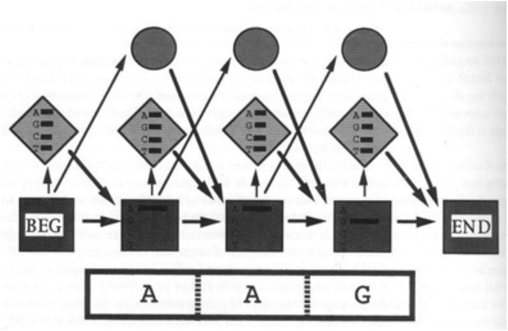
Hidden Markov Model

- Also has the Markov property.
- Some of the state or transition probabilities information is missing.
- The process emits sequences of results.
- The emission probabilities is the probability of each outcome in a given state.
- The model is trained so that the training set is the most likely outcome for the model

Training and Testing the HMM

- The parameters of the model are fit on a training set, ie., the parameters are chosen so that the training set is the most likely outcome for the model.
- A test set is used to make sure the model is well-trained.
- If so, the model can be used on new data.

HMM of *E. Coli* Gene



match

insert

delete

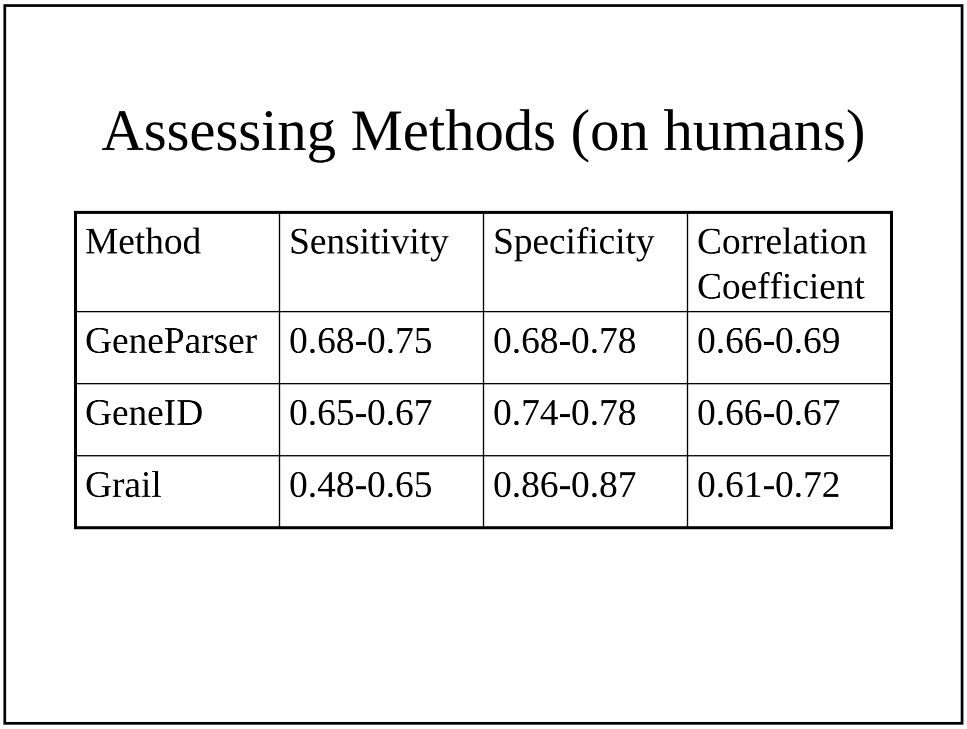
- HMM for finding the most probable set of genes in *E. coli* gene sequences of unknown gene composition.
- A similar model exists for each of the 61 codons

HMM of *E. Coli* Genes

- Assumes that there is no relationship each codon and codons used later in the sequence.
- This assumption works, however, analysis of sequential codons in a gene have shown that some pairs are found at greater/lesser frequencies than would occur at random.
- GeneMark.HMM uses sequence information from the previous 5 bases instead of the previous 2 bases.

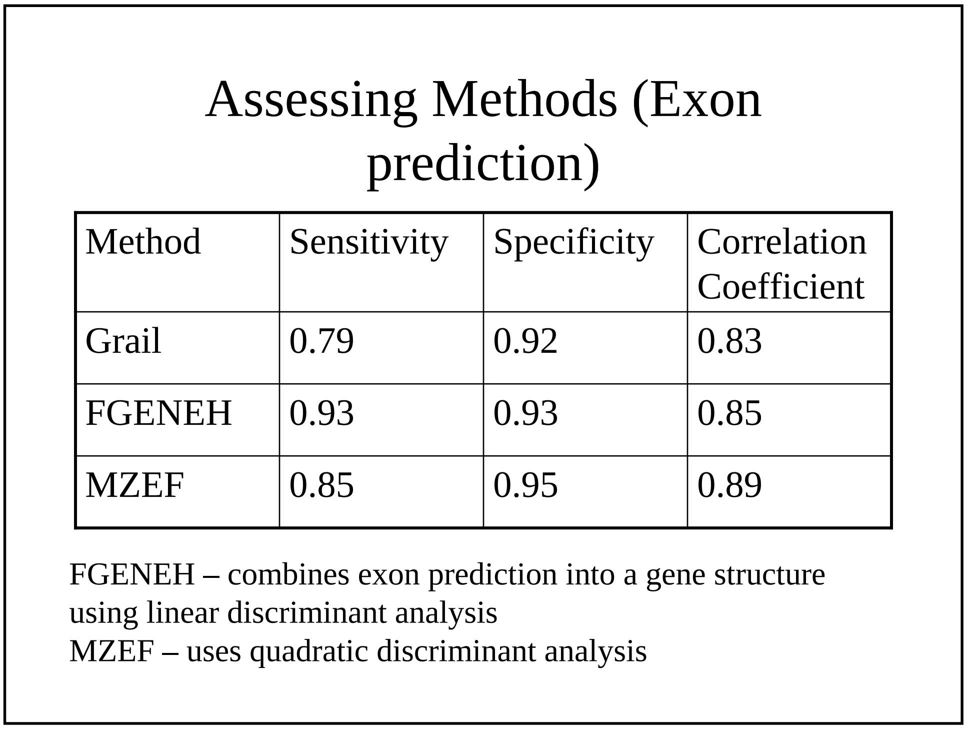
Assessing Methods

- Take a set of know genes and test method for true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).
- Use these to calculate
  - Specificity = TP/(TP+FN)
  - Sensitivity = TN/(TN+FP)
  - Correlation coefficient = [(TP)(TN)-(FP)(FN)]/ SQRT[(AN)(TP+FP)(AP)(TN+FN)]



Assessing Methods (on humans)

Method	Sensitivity	Specificity	Correlation Coefficient
GeneParser	0.68-0.75	0.68-0.78	0.66-0.69
GeneID	0.65-0.67	0.74-0.78	0.66-0.67
Grail	0.48-0.65	0.86-0.87	0.61-0.72



Assessing Methods (Exon prediction)

Method	Sensitivity	Specificity	Correlation Coefficient
Grail	0.79	0.92	0.83
FGENEH	0.93	0.93	0.85
MZEF	0.85	0.95	0.89

FGENEH – combines exon prediction into a gene structure using linear discriminant analysis
MZEF – uses quadratic discriminant analysis