If the data does not come to R, R must go to the data

Olga Kalinina

Helmholtz Institute for Pharmaceutical Research Saarland, Saarland University

FOSDEM PGDay 2019

Who am I?

Who am I?

- Bioinformatics = computational biology

Who am I?

- Bioinformatics = computational biology
  - Analysis of data to gain new biological insights

Who am I?

- Bioinformatics = computational biology
  - Analysis of data to gain new biological insights
  - Molecular biology

Who am I?

- Bioinformatics = computational biology
  - Analysis of data to gain new biological insights
  - Molecular biology
- Head of research group for drug bioinformatics at Helmholtz Institute for Pharmaceutical Research Saarland

Who am I?

- Bioinformatics = computational biology
  - Analysis of data to gain new biological insights
  - Molecular biology
- Head of research group for drug bioinformatics at Helmholtz Institute for Pharmaceutical Research Saarland
  - Find new bioactive compounds

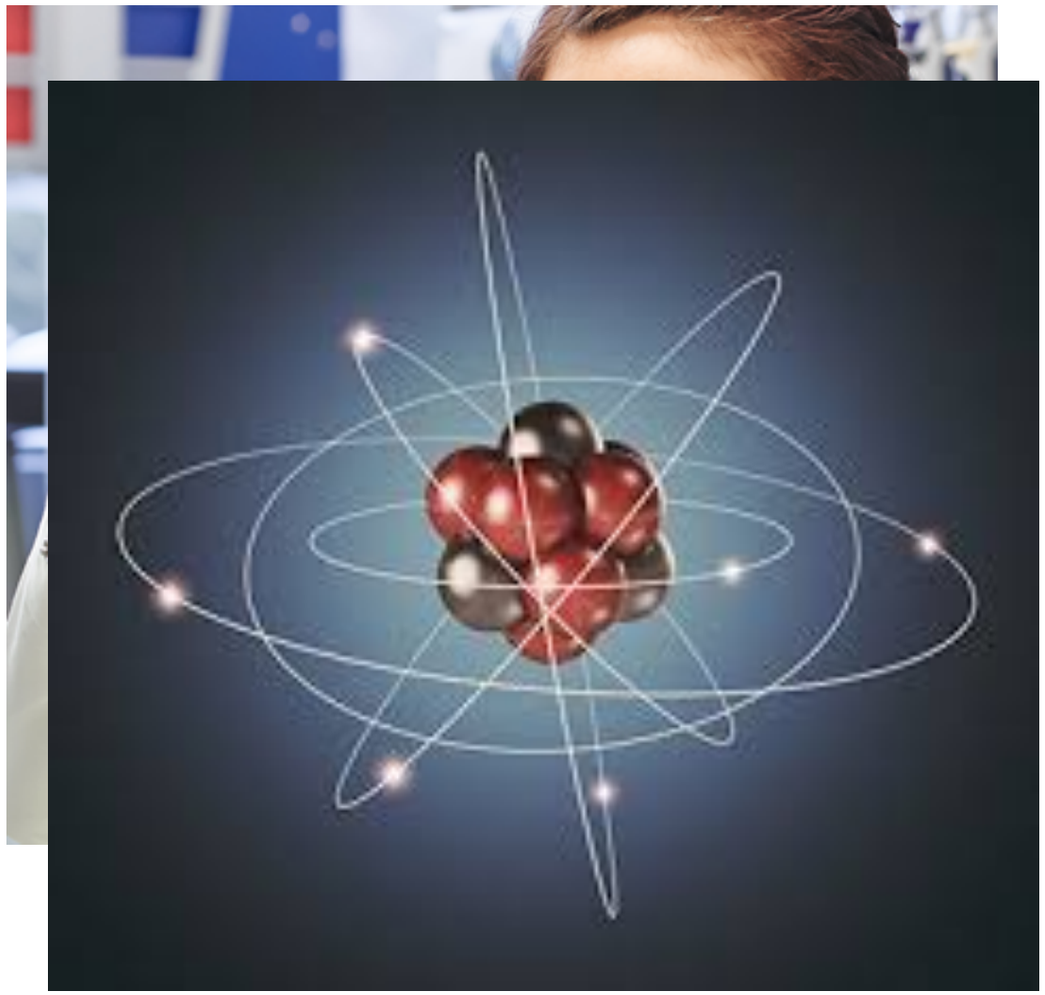
Data in (life) sciences

Data in (life) sciences

Data in (life) sciences

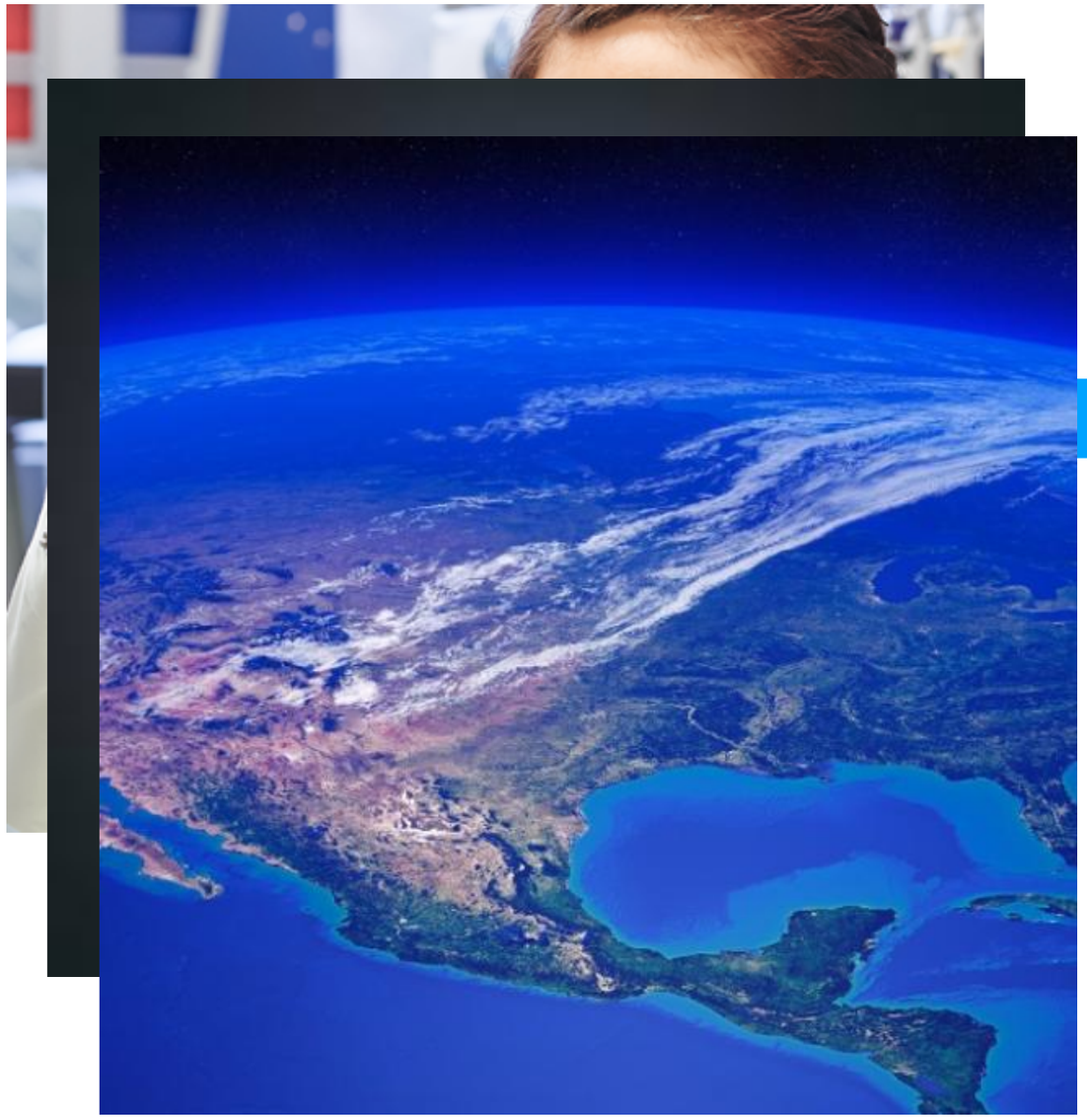
#chr	to1	ref	alt	GeneSymbol	ClinicalSignificance	ReviewStatus	PhenotypeList	uniprot_ac	uniprot_pos	aa1	aa2
chr1	949608	G	A	ISG15	Benign	criteria provided, single submitter	not specified	P05161 83	S N	benign	3rt3
chr1	955563	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	4 R	P	benign
chr1	955596	C	G	AGRN	Benign	criteria provided, single submitter	not provided	O00468-6	15 P	R	benign
chr1	955601	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	17 L	F	possibly
chr1	957605	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	76 G		
chr1	957693	A	T	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	105 N		
chr1	976577	T	C	AGRN	Benign	no assertion criteria provided	not specified	O00468-6	251 V	A	probably damagin
chr1	976598	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	258 T	I	probably
chr1	976963	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	353 Q	R	benign
chr1	977028	G	T	AGRN	Benign	criteria provided, multiple submitters,	no conflicts	not provided;not specified		O00468-6	
chr1	978628	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	465 P	L	benign
chr1	978762	G	A	AGRN	Benign	criteria provided, multiple submitters,	no conflicts	not specified	O00468-6	510 G	
chr1	978974	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	554 V	M	probably
chr1	979310	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	636 G	S	probably
chr1	979748	A	T	AGRN	Benign	criteria provided, multiple submitters,	no conflicts	not specified	O00468-6	728 E	
chr1	980552	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	756 A	T	possibly
chr1	980840	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	825 R	C	probably
chr1	980868	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	834 R	Q	probably
chr1	981131	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	852 Q	R	benign
chr1	981226	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	884 R	C	probably
chr1	981353	C	T	AGRN	Benign	criteria provided, single submitter	not provided	O00468-6	897 A	V	possibly
chr1	981942	C	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1026 T	N	probably
chr1	982213	G	C	AGRN	Benign	criteria provided, multiple submitters,	no conflicts	not provided;not specified		O00468-6	
chr1	982722	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1135 Q	R	probably
chr1	983221	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1233 R	W	benign
chr1	983243	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1240 P	L	probably
chr1	983506	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1289 P	L	benign
chr1	983604	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1322 R	W	probably
chr1	984261	G	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1374 V	L	benign
chr1	984426	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1429 R	C	benign
chr1	984669	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1451 P	L	benign
chr1	984971	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1514 A	T	benign
chr1	985070	G	A	AGRN	Benign	criteria provided, multiple submitters,	no conflicts	not specified	O00468-6	1547 E	
chr1	985126	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1565 Q	H	benign
chr1	985407	C	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1623 F	L	benign
chr1	985826	G	A	AGRN	Benign	criteria provided, multiple submitters,	no conflicts	not specified	O00468-6	1666 V	
chr1	985853	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	1675 G		
chr1	985955	G	C	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome;Myasthenic syndrome, congenital,				
chr1	986143	G	T	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome;Myasthenic syndrome, congenital,				
chr1	986165	G	A	AGRN	Benign	criteria provided, multiple submitters,	no conflicts	not specified	O00468-6	1734 R	
chr1	986849	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1796 R	H	benign
chr1	987116	G	A	AGRN	Benign	criteria provided, single submitter	Myasthenic syndrome, congenital, 8	O00468-6	1858		
chr1	987155	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	1871 G		
chr1	987159	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1872 R	Q	benign
chr1	987191	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1883 E	K	benign
chr1	989207	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1909 S	T	benign
chr1	989224	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1915 R	W	probably
chr1	990213	C	T	AGRN	Benign	no assertion criteria provided	not specified	O00468-6	1997 P	L	benign ?
chr1	990242	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	2007 K	E	probably
chr1	1167674	C	T	B3GALT6	Pathogenic	no assertion criteria provided	Ehlers-Danlos syndrome, progeroid type, 2	Q96L58 6			
chr1	1167675	G	A	B3GALT6	Benign	no assertion criteria provided	not specified	Q96L58 6	R Q	possibly damaging	
chr1	1167680	T	G	B3GALT6	Benign	criteria provided, single submitter	not specified	Q96L58 8	W G	benign ?	

Data in (life) sciences



#chr	to1	ref	alt	GeneSymbol	ClinicalSignificance	ReviewStatus	PhenotypeList	uniprot_ac	uniprot_pos	aa1	aa2
chr1	949608	G	A	ISG15	Benign	criteria provided, single submitter	not specified	P05161 83	S N	benign	3rt3
chr1	955563	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	4 R	P	benign
chr1	955596	C	G	AGRN	Benign	criteria provided, single submitter	not provided	O00468-6	15 P	R	benign
chr1	955601	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	17 L	F	possibly
chr1	957605	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	76 G		
chr1	957693	A	T	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	105 N		
chr1	976577	T	C	AGRN	Benign	no assertion criteria provided	not specified	O00468-6	251 V A	probably damagin	
chr1	976598	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	258 T	I	probably
chr1	976963	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	353 Q	R	benign
chr1	977028	G	T	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not provided;not specified	O00468-6			
chr1	978628	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	465 P	L	benign
chr1	978762	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	510 G		
chr1	978974	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	554 V	M	probably
chr1	979310	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	636 G	S	probably
chr1	979748	A	T	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	728 E		
chr1	980552	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	756 A	T	possibly
chr1	980840	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	825 R	C	probably
chr1	980868	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	834 R	Q	probably
chr1	981131	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	852 Q	R	benign
chr1	981226	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	884 R	C	probably
chr1	981353	C	T	AGRN	Benign	criteria provided, single submitter	not provided	O00468-6	897 A	V	possibly
chr1	981942	C	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1026 T	N	probably
chr1	982213	G	C	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not provided;not specified	O00468-6			
chr1	982722	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1135 Q	R	probably
chr1	983221	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1233 R	W	benign
chr1	983243	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1240 P	L	probably
chr1	983506	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1289 P	L	benign
chr1	983604	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1322 R	W	probably
chr1	984261	G	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1374 V	L	benign
chr1	984426	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1429 R	C	benign
chr1	984669	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1451 P	L	benign
chr1	984971	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1514 A	T	benign
chr1	985070	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1547 E		
chr1	985126	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1565 Q	H	benign
chr1	985407	C	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1623 F	L	benign
chr1	985826	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1666 V		
chr1	985853	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	1675 G		
chr1	985955	G	C	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome;Myasthenic syndrome, congenital,				
chr1	986143	G	T	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome;Myasthenic syndrome, congenital,				
chr1	986165	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1734 R		
chr1	986849	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1796 R	H	benign
chr1	987116	G	A	AGRN	Benign	criteria provided, single submitter	Myasthenic syndrome, congenital, 8	O00468-6	1858		
chr1	987155	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	1871 G		
chr1	987159	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1872 R	Q	benign
chr1	987191	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1883 E	K	benign
chr1	989207	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1909 S	T	benign
chr1	989224	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1915 R	W	probably
chr1	990213	C	T	AGRN	Benign	no assertion criteria provided	not specified	O00468-6	1997 P	L	benign ?
chr1	990242	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	2007 K	E	probably
chr1	1167674	C	T	B3GALT6	Pathogenic	no assertion criteria provided	Ehlers-Danlos syndrome, progeroid type, 2	Q96L58 6	R Q	possibly damaging	
chr1	1167675	G	A	B3GALT6	Benign	no assertion criteria provided	not specified	Q96L58 6	R Q	possibly damaging	
chr1	1167680	T	G	B3GALT6	Benign	criteria provided, single submitter	not specified	Q96L58 8	W G	benign ?	

Data in (life) sciences



#chr	to1	ref	alt	GeneSymbol	ClinicalSignificance	ReviewStatus	PhenotypeList	uniprot_ac	uniprot_pos	aa1	aa2
chr1	949608	G	A	ISG15	Benign	criteria provided, single submitter	not specified	P05161 83	S N	benign	3rt3
chr1	955563	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	4 R	P	benign
chr1	955596	C	G	AGRN	Benign	criteria provided, single submitter	not provided	O00468-6	15 P	R	benign
chr1	955601	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	17 L	F	possibly
chr1	957605	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	76 G		
chr1	957693	A	T	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	105 N		
chr1	976577	T	C	AGRN	Benign	no assertion criteria provided	not specified	O00468-6	251 V A	probably damagin	
chr1	976598	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	258 T	I	probably
chr1	976963	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	353 Q	R	benign
chr1	977028	G	T	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not provided;not specified	O00468-6			
chr1	978628	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	465 P	L	benign
chr1	978762	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	510 G		
chr1	978974	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	554 V	M	probably
chr1	979310	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	636 G	S	probably
chr1	979748	A	T	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	728 E		
chr1	980552	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	756 A	T	possibly
chr1	980840	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	825 R	C	probably
chr1	980868	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	834 R	Q	probably
chr1	981131	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	852 Q	R	benign
chr1	981226	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	884 R	C	probably
chr1	981353	C	T	AGRN	Benign	criteria provided, single submitter	not provided	O00468-6	897 A	V	possibly
chr1	981942	C	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1026 T	N	probably
chr1	982213	G	C	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not provided;not specified	O00468-6			
chr1	982722	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1135 Q	R	probably
chr1	983221	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1233 R	W	benign
chr1	983243	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1240 P	L	probably
chr1	983506	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1289 P	L	benign
chr1	983604	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1322 R	W	probably
chr1	984261	G	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1374 V	L	benign
chr1	984426	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1429 R	C	benign
chr1	984669	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1451 P	L	benign
chr1	984971	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1514 A	T	benign
chr1	985070	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1547 E		
chr1	985126	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1565 Q	H	benign
chr1	985407	C	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1623 F	L	benign
chr1	985826	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1666 V		
chr1	985853	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	1675 G		
chr1	985955	G	C	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome;Myasthenic syndrome, congenital,				
chr1	986143	G	T	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome;Myasthenic syndrome, congenital,				
chr1	986165	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1734 R		
chr1	986849	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1796 R	H	benign
chr1	987116	G	A	AGRN	Benign	criteria provided, single submitter	Myasthenic syndrome, congenital, 8	O00468-6	1858		
chr1	987155	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	1871 G		
chr1	987159	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1872 R	Q	benign
chr1	987191	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1883 E	K	benign
chr1	989207	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1909 S	T	benign
chr1	989224	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1915 R	W	probably
chr1	990213	C	T	AGRN	Benign	no assertion criteria provided	not specified	O00468-6	1997 P	L	benign ?
chr1	990242	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	2007 K	E	probably
chr1	1167674	C	T	B3GALT6	Pathogenic	no assertion criteria provided	Ehlers-Danlos syndrome, progeroid type, 2	Q96L58 6	R Q	possibly damaging	
chr1	1167675	G	A	B3GALT6	Benign	no assertion criteria provided	not specified	Q96L58 6	R Q	possibly damaging	
chr1	1167680	T	G	B3GALT6	Benign	criteria provided, single submitter	not specified	Q96L58 8	W G	benign ?	

Data in (life) sciences

#chr	to1	ref	alt	GeneSymbol	ClinicalSignificance	ReviewStatus	PhenotypeList	uniprot_ac	uniprot_pos	aa1	aa2
chr1	949608	G	A	ISG15	Benign	criteria provided, single submitter	not specified	P05161 83	S	N	benign 3rt3
chr1	955563	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	4	R	P benign
chr1	955596	C	G	AGRN	Benign	criteria provided, single submitter	not provided	O00468-6	15	P	R benign
chr1	955601	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	17	L	F possibly
chr1	957605	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	76	G	
chr1	957693	A	T	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	105	N	
chr1	976577	T	C	AGRN	Benign	no assertion criteria provided	not specified O00468-6	251	V	A	probably damagin
chr1	976598	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	258	T	I probably
chr1	976963	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	353	Q	R benign
chr1	977028	G	T	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not provided;not specified	O00468-6			
chr1	978628	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	465	P	L benign
chr1	978762	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	510	G	
chr1	978974	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	554	V	M probably
chr1	979310	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	636	G	S probably
chr1	979748	A	T	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	728	E	
chr1	980552	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	756	A	T possibly
chr1	980840	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	825	R	C probably
chr1	980868	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	834	R	Q probably
chr1	981131	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	852	Q	R benign
chr1	981226	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	884	R	C probably
chr1	981353	C	T	AGRN	Benign	criteria provided, single submitter	not provided	O00468-6	897	A	V possibly
chr1	981942	C	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1026	T	N probably
chr1	982213	G	C	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not provided;not specified	O00468-6			
chr1	982722	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1135	Q	R probably
chr1	983221	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1233	R	W benign
chr1	983243	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1240	P	L probably
chr1	983506	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1289	P	L benign
chr1	983604	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1322	R	W probably
chr1	984261	G	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1374	V	L benign
chr1	984426	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1429	R	C benign
chr1	984669	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1451	P	L benign
chr1	984971	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1514	A	T benign
chr1	985070	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1547	E	
chr1	985126	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1565	Q	H benign
chr1	985407	C	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1623	F	L benign
chr1	985826	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1666	V	
chr1	985853	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	1675	G	
chr1	985955	G	C	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome;Myasthenic syndrome, congenital,				
chr1	986143	G	T	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome;Myasthenic syndrome, congenital,				
chr1	986165	G	A	AGRN	Benign	criteria provided, multiple submitters, no conflicts	not specified	O00468-6	1734	R	
chr1	986849	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1796	R	H benign
chr1	987116	G	A	AGRN	Benign	criteria provided, single submitter	Myasthenic syndrome, congenital, 8	O00468-6	1858		
chr1	987155	G	A	AGRN	Pathogenic	no assertion criteria provided	Congenital myasthenic syndrome	O00468-6	1871	G	
chr1	987159	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1872	R	Q benign
chr1	987191	G	A	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1883	E	K benign
chr1	989207	G	C	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1909	S	T benign
chr1	989224	C	T	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	1915	R	W probably
chr1	990213	C	T	AGRN	Benign	no assertion criteria provided	not specified O00468-6	1997	P	L	benign ?
chr1	990242	A	G	AGRN	Benign	criteria provided, single submitter	not specified	O00468-6	2007	K	E probably
chr1	1167674	C	T	B3GALT6	Pathogenic	no assertion criteria provided	Ehlers-Danlos syndrome, progeroid type, 2				Q96L58 6
chr1	1167675	G	A	B3GALT6	Benign	no assertion criteria provided	not specified Q96L58 6	R	Q	possibly damaging	
chr1	1167680	T	G	B3GALT6	Benign	criteria provided, single submitter	not specified Q96L58 8	W	G	benign ?	

Where does the data come from?

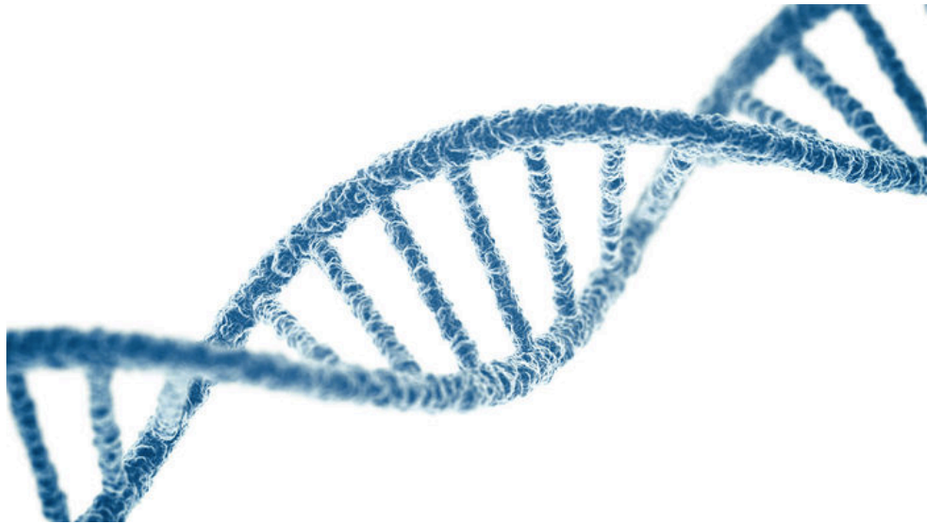
Where does the data come from?

- Experiment

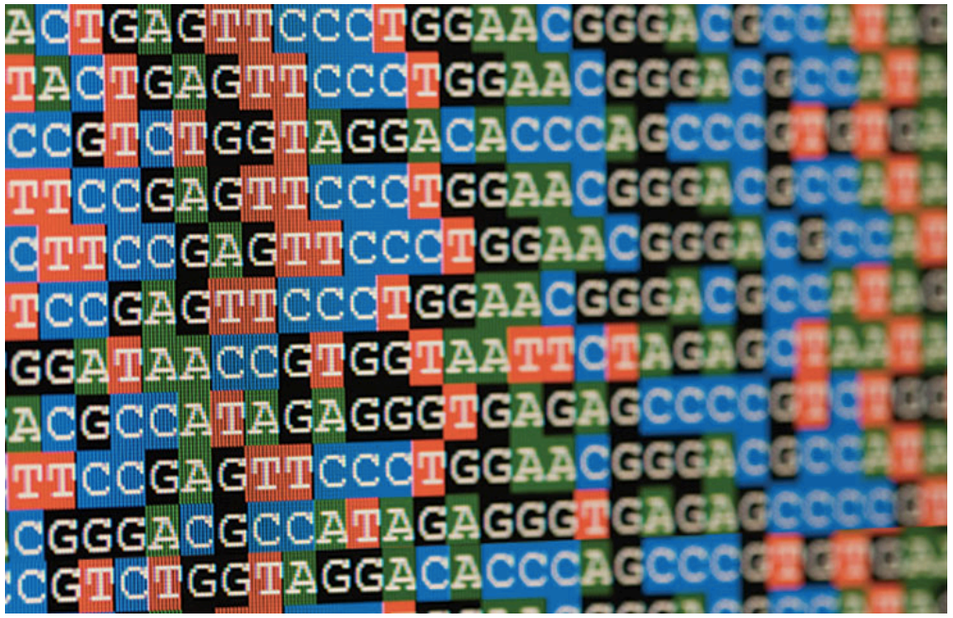
Where does the data come from?

- Experiment
  - Genome sequencing

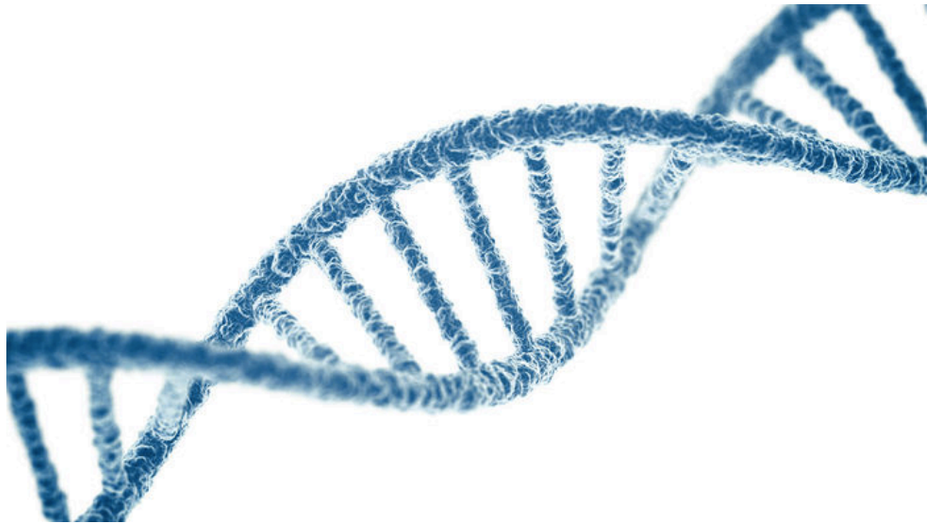
Where does the data come from?



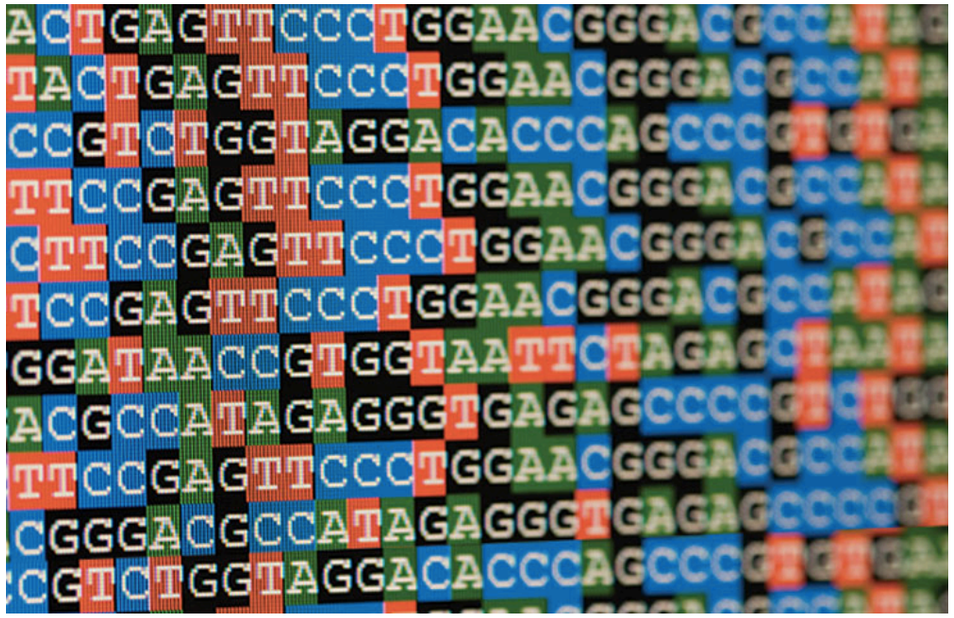
- Experiment
  - Genome sequencing



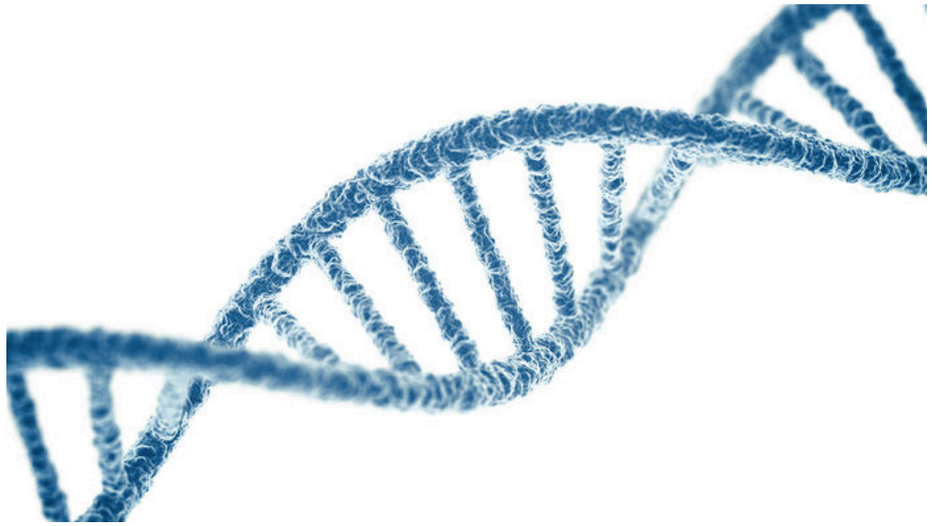
Where does the data come from?



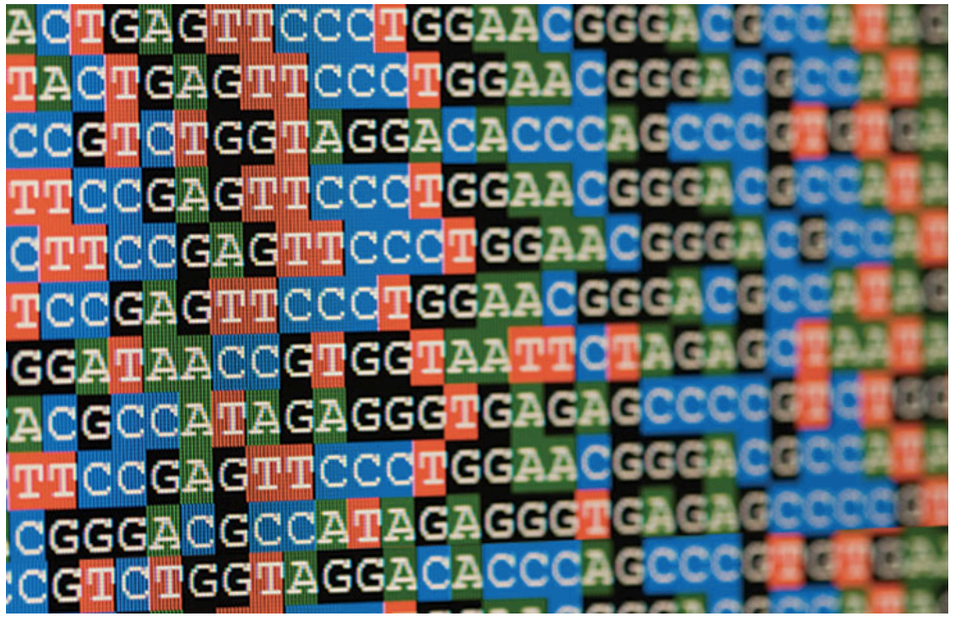
- Experiment
  - Genome sequencing
    - => ~4×10^{12} bp



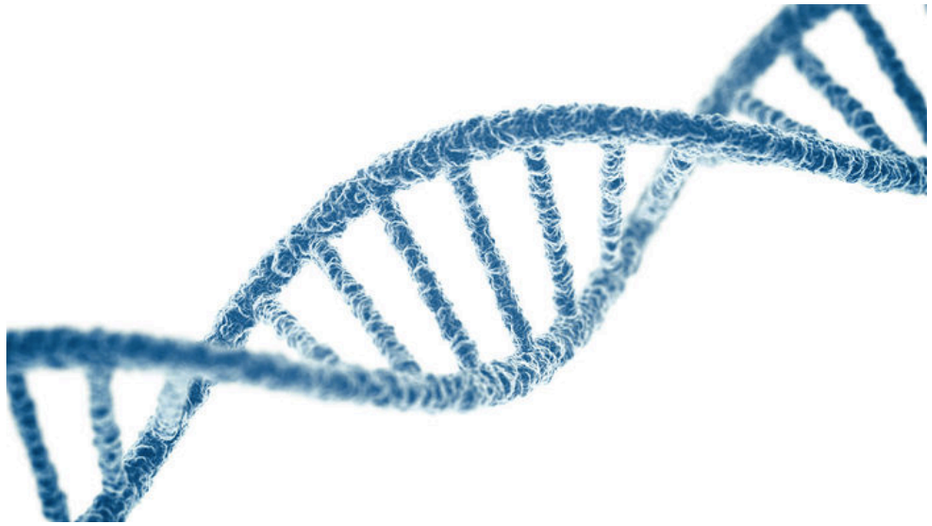
Where does the data come from?



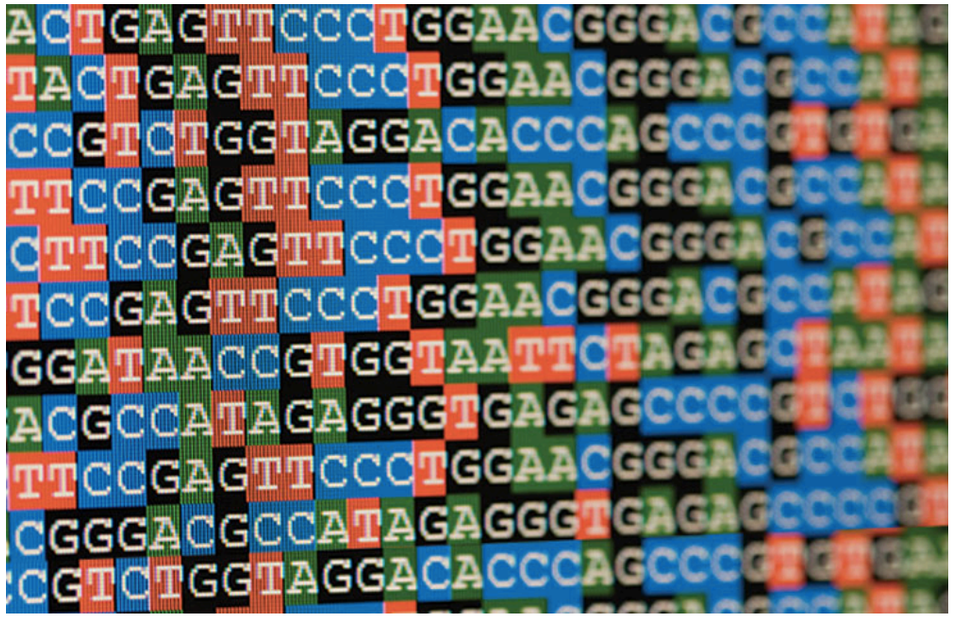
- Experiment
  - Genome sequencing
    - => ~4×10^{12} bp
  - Other types of experiment



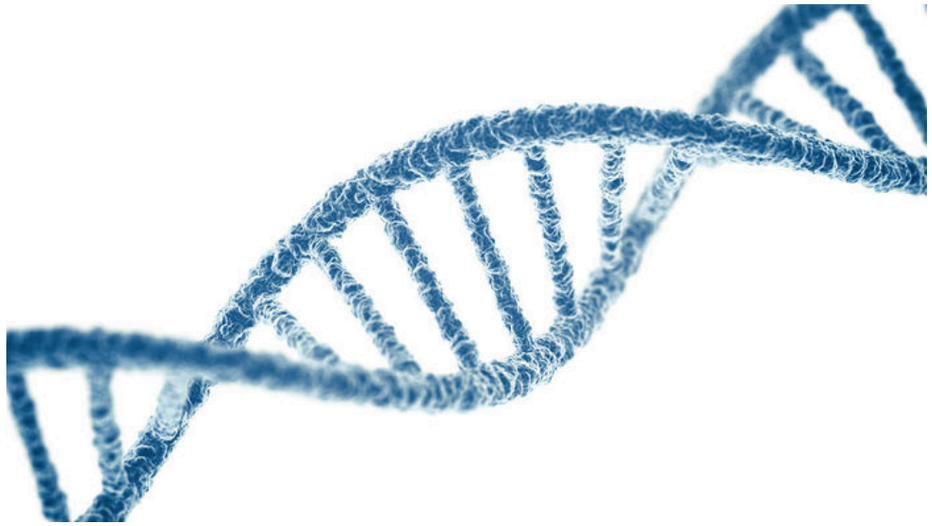
Where does the data come from?



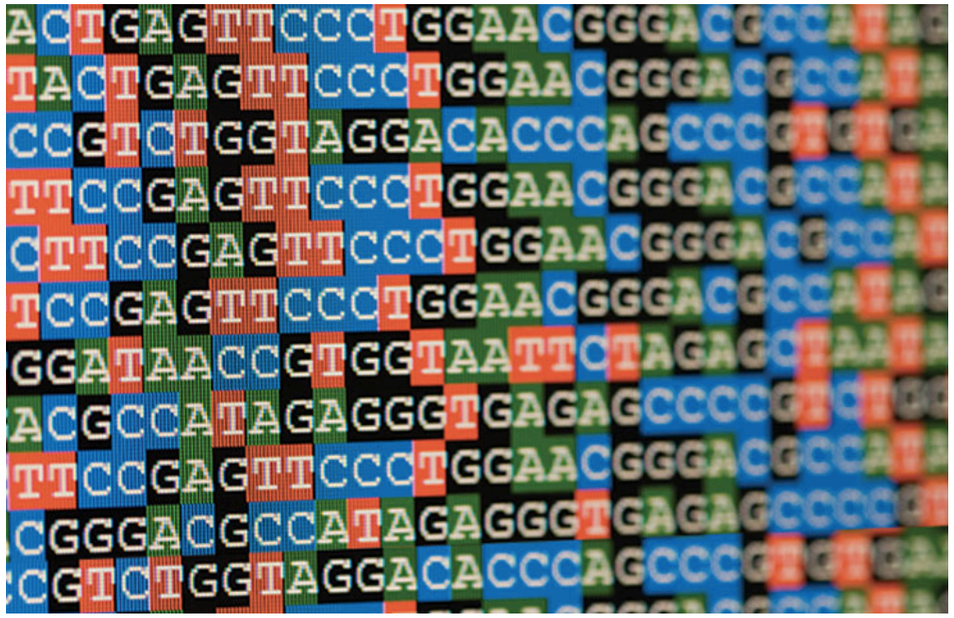
- Experiment
  - Genome sequencing
    - => ~4×10^{12} bp
  - Other types of experiment
    - Determination of protein 3D structure



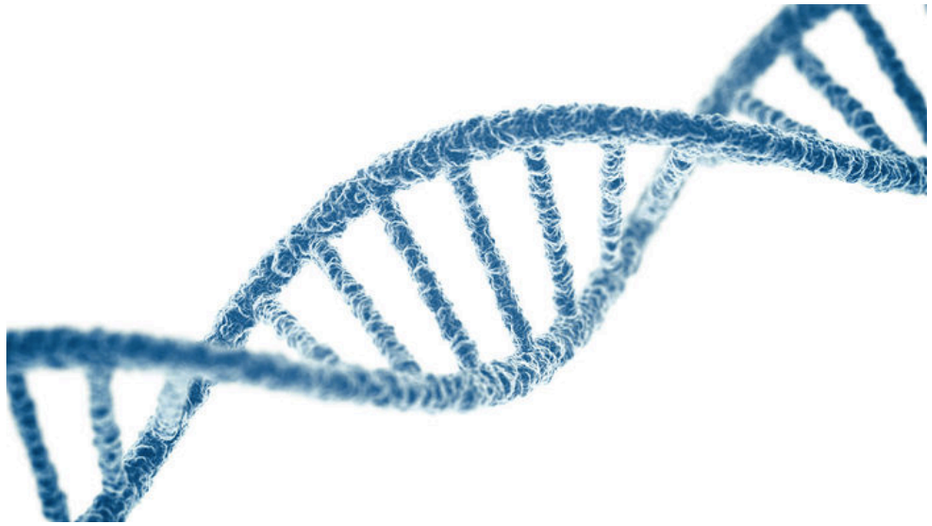
Where does the data come from?



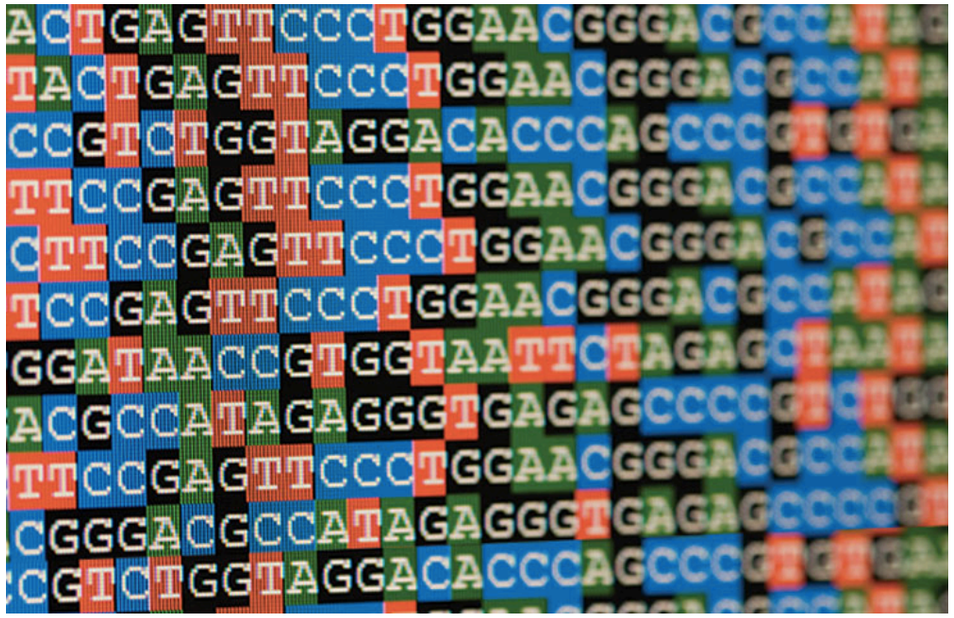
- Experiment
  - Genome sequencing
    - => ~4×10^{12} bp
  - Other types of experiment
    - Determination of protein 3D structure
    - Gene expression



Where does the data come from?



- Experiment
  - Genome sequencing
    - => ~4×10^{12} bp
  - Other types of experiment
    - Determination of protein 3D structure
    - Gene expression
    - Computational predictions



How BIG is the data?

How BIG is the data?

- All DNA sequences: ~4×10^{12} bp = ~9 GB + metadata

How BIG is the data?

- All DNA sequences: ~4×10^{12} bp = ~9 GB + metadata
- In this talk:

How BIG is the data?

- All DNA sequences: ~4×10^{12} bp = ~9 GB + metadata
- In this talk:
  - Clinically relevant mutations: 13 MB = 84,426 rows

How BIG is the data?

- All DNA sequences: ~4×10^{12} bp = ~9 GB + metadata
- In this talk:
  - Clinically relevant mutations: 13 MB = 84,426 rows
  - All human proteins + annotations: 1.9 GB = 23,095,049 rows

How BIG is the data?

- All DNA sequences: ~4×10^{12} bp = ~9 GB + metadata
- In this talk:
  - Clinically relevant mutations: 13 MB = 84,426 rows
  - All human proteins + annotations: 1.9 GB = 23,095,049 rows
  - (Cross-references from human proteins to other data sources: 147 MB = 6,026,631 rows)

Typical data analysis pipeline

Typical data analysis pipeline

Experiment (up to TBs of data)

Typical data analysis pipeline

Experiment (up to TBs of data)

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

↓

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

↓

Store in a DB

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

↓

Store in a DB

↓

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

↓

Store in a DB

↓

Select relevant data

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

↓

Store in a DB

↓

Select relevant data

↓

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

↓

Store in a DB

↓

Select relevant data

↓

Write to disc (text files, MBs to GBs)

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

↓

Store in a DB

↓

Select relevant data

↓

Write to disc (text files, MBs to GBs)

↓

Typical data analysis pipeline

Experiment (up to TBs of data)

↓

Initial data processing, cross-referencing

↓

Store in a DB

↓

Select relevant data

↓

Write to disc (text files, MBs to GBs)

↓

Analyze with dedicated statistical software (Python, SAS, R), typically in RAM

R programming language

R programming language

- Free software environment for statistical computing and graphics

R programming language

- Free software environment for statistical computing and graphics

- Introduced in 1993

R programming language

- Free software environment for statistical computing and graphics

- Introduced in 1993

- Multi-paradigm, including **array**: many generalized functions for multi-dimensional data (vectors, matrices, …)

R programming language

- Free software environment for statistical computing and graphics

- Introduced in 1993

- Multi-paradigm, including **array**: many generalized functions for multi-dimensional data (vectors, matrices, …)

- R project: https://www.r-project.org/

R programming language

- Free software environment for statistical computing and graphics

- Introduced in 1993

- Multi-paradigm, including **array**: many generalized functions for multi-dimensional data (vectors, matrices, …)

- R project: https://www.r-project.org/

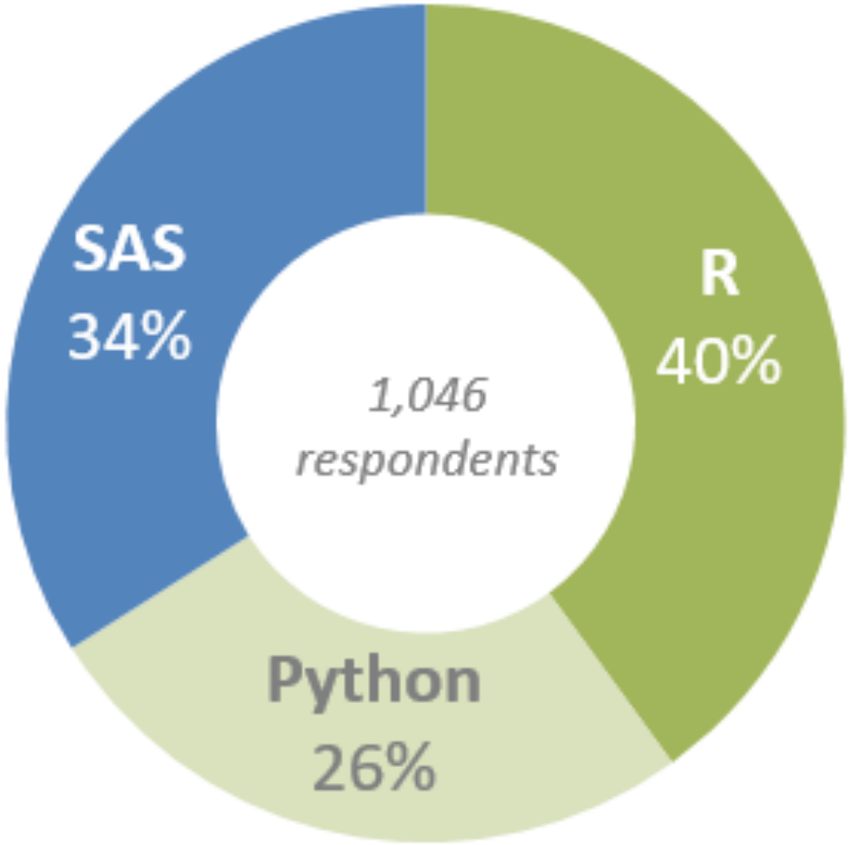
- CRAN — 13,626 packages for various types of analysis: https://cran.r-project.org/

R

- R is still widely used, especially in academia

R

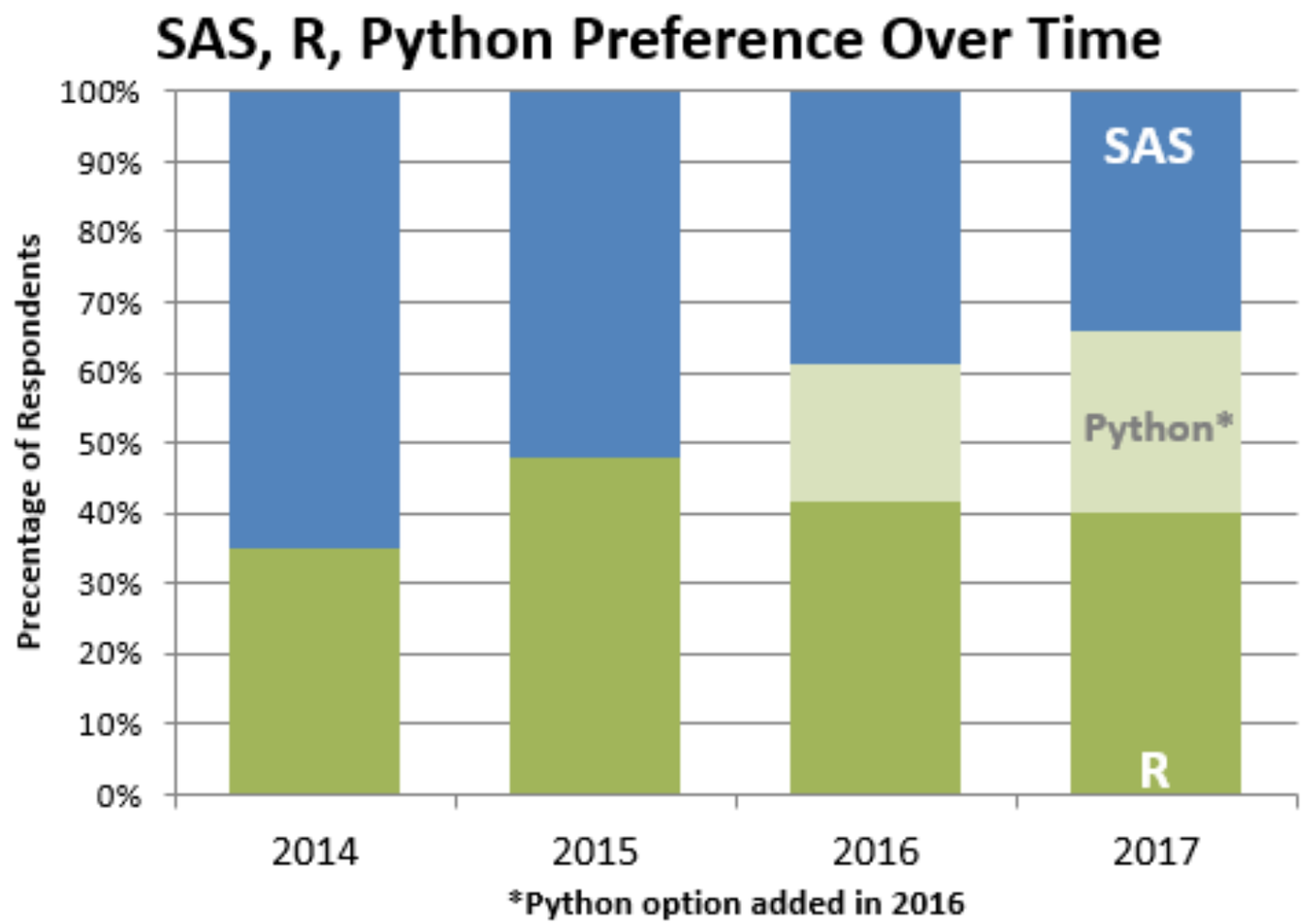
- R is still widely used, especially in academia



Source: https://www.burtchworks.com/2017/06/19/2017-sas-r-python-flash-survey-results/

R

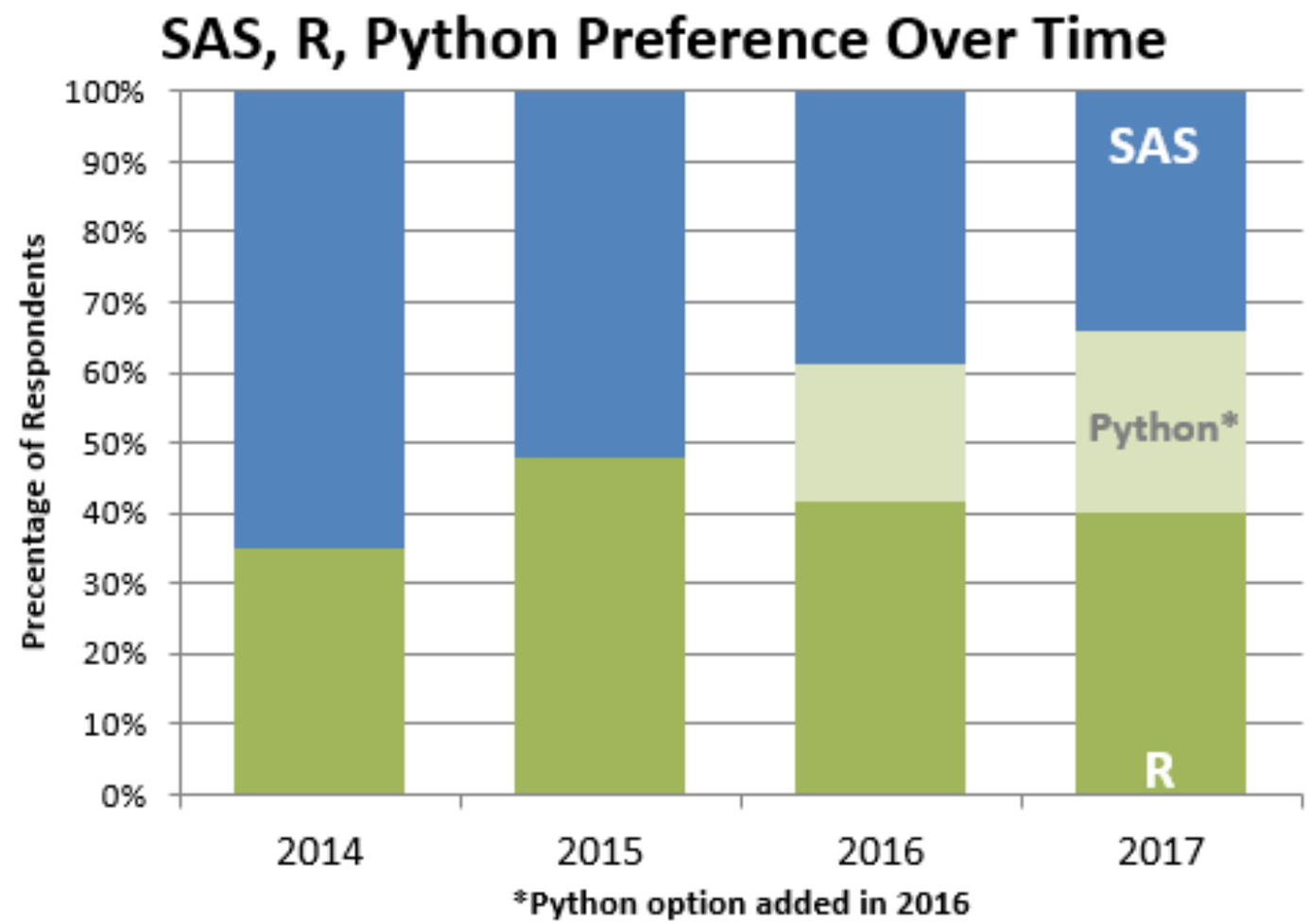
- R is still widely used, especially in academia

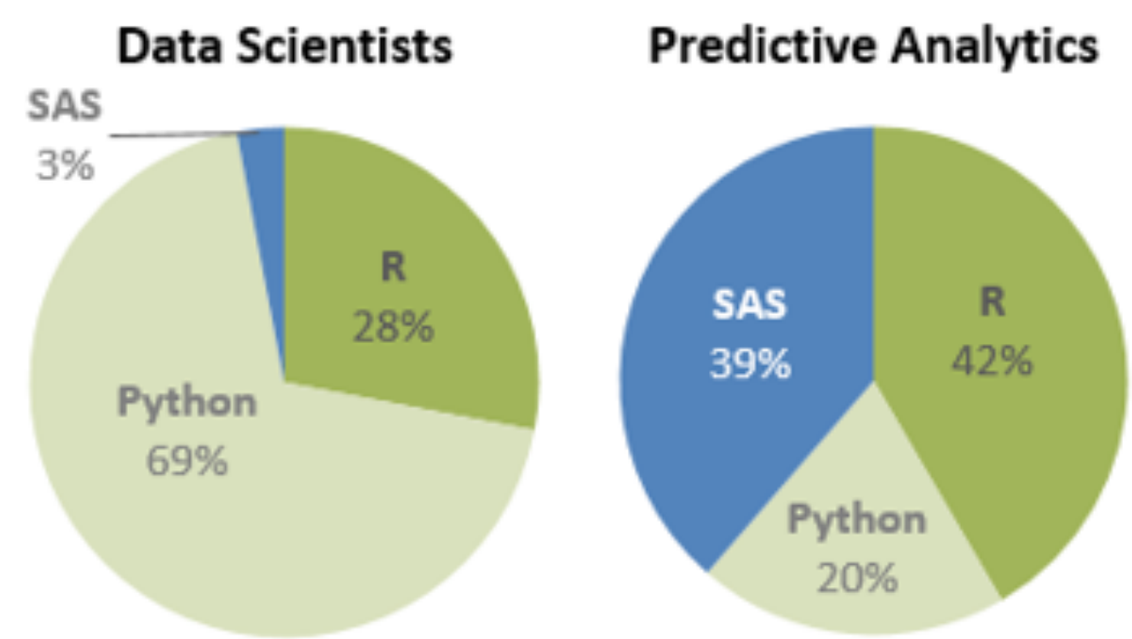


Source: https://www.burtchworks.com/2017/06/19/2017-sas-r-python-flash-survey-results/

R

- R is still widely used, especially in academia

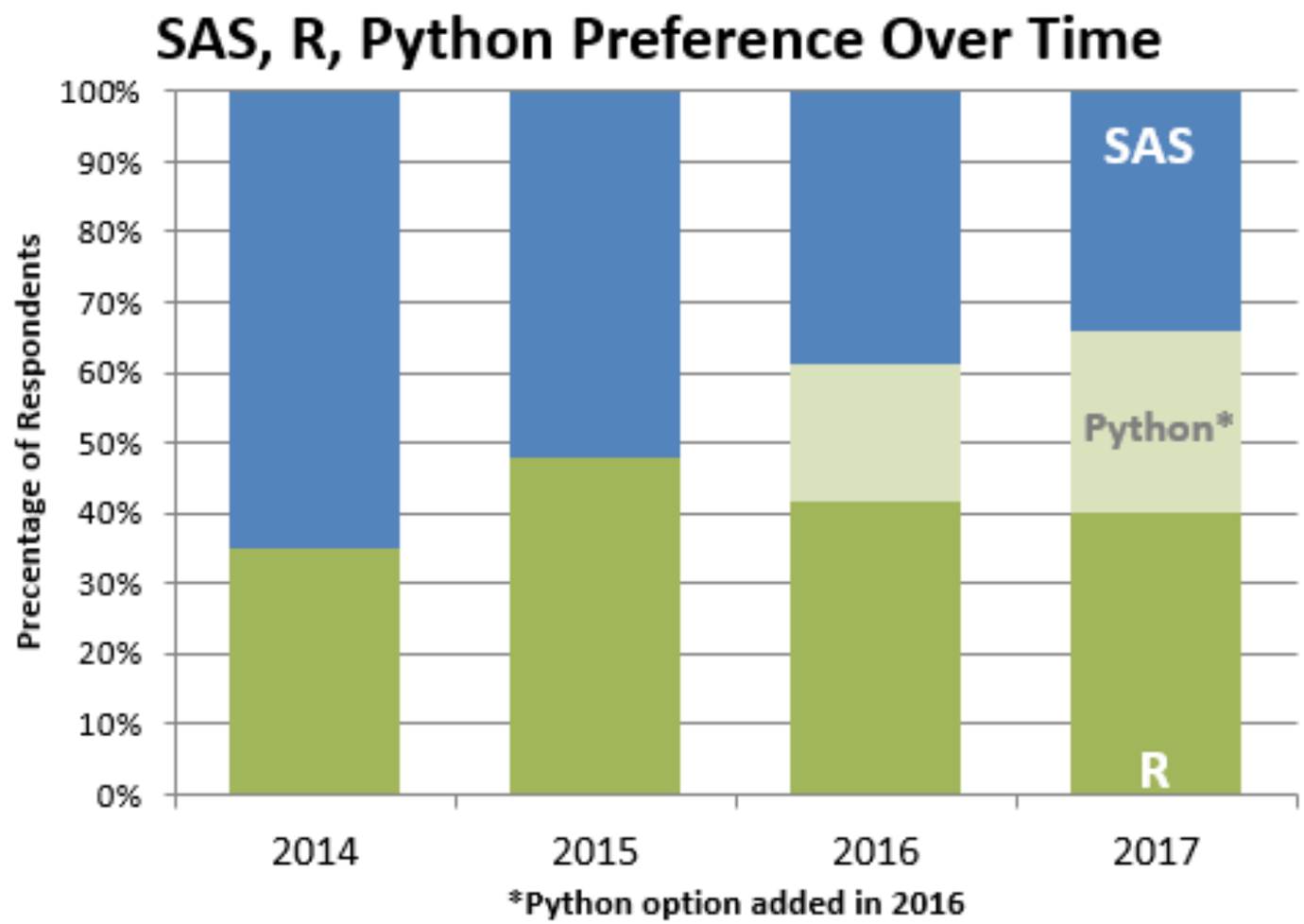


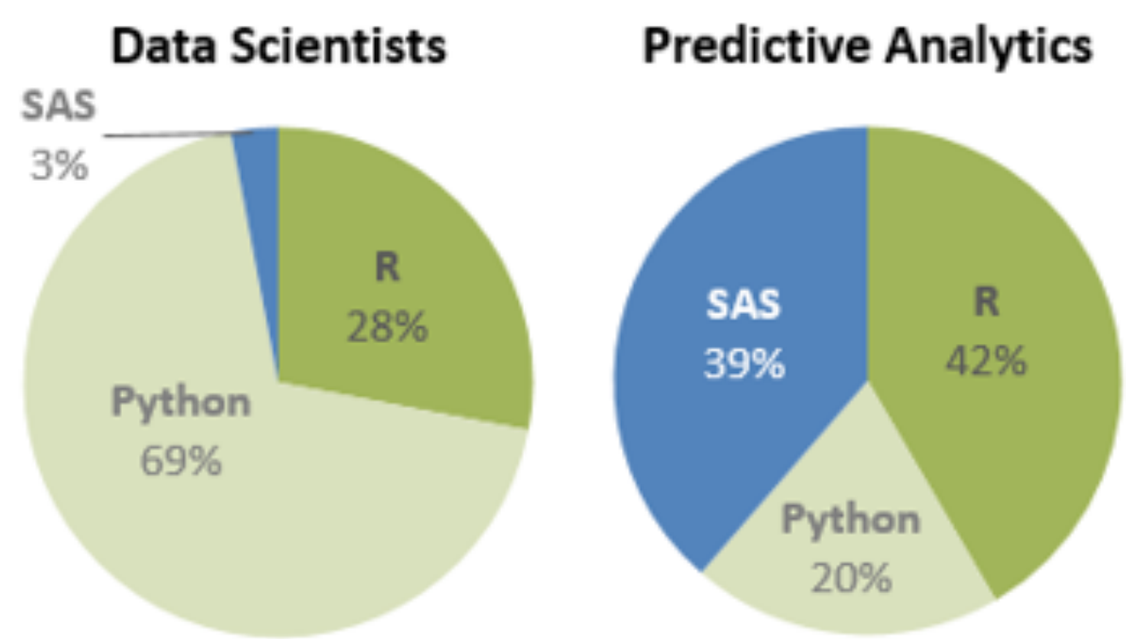


Source: https://www.burtchworks.com/2017/06/19/2017-sas-r-python-flash-survey-results/

R

- R is still widely used, especially in academia
- R is very well suited to do statistical / machine learning

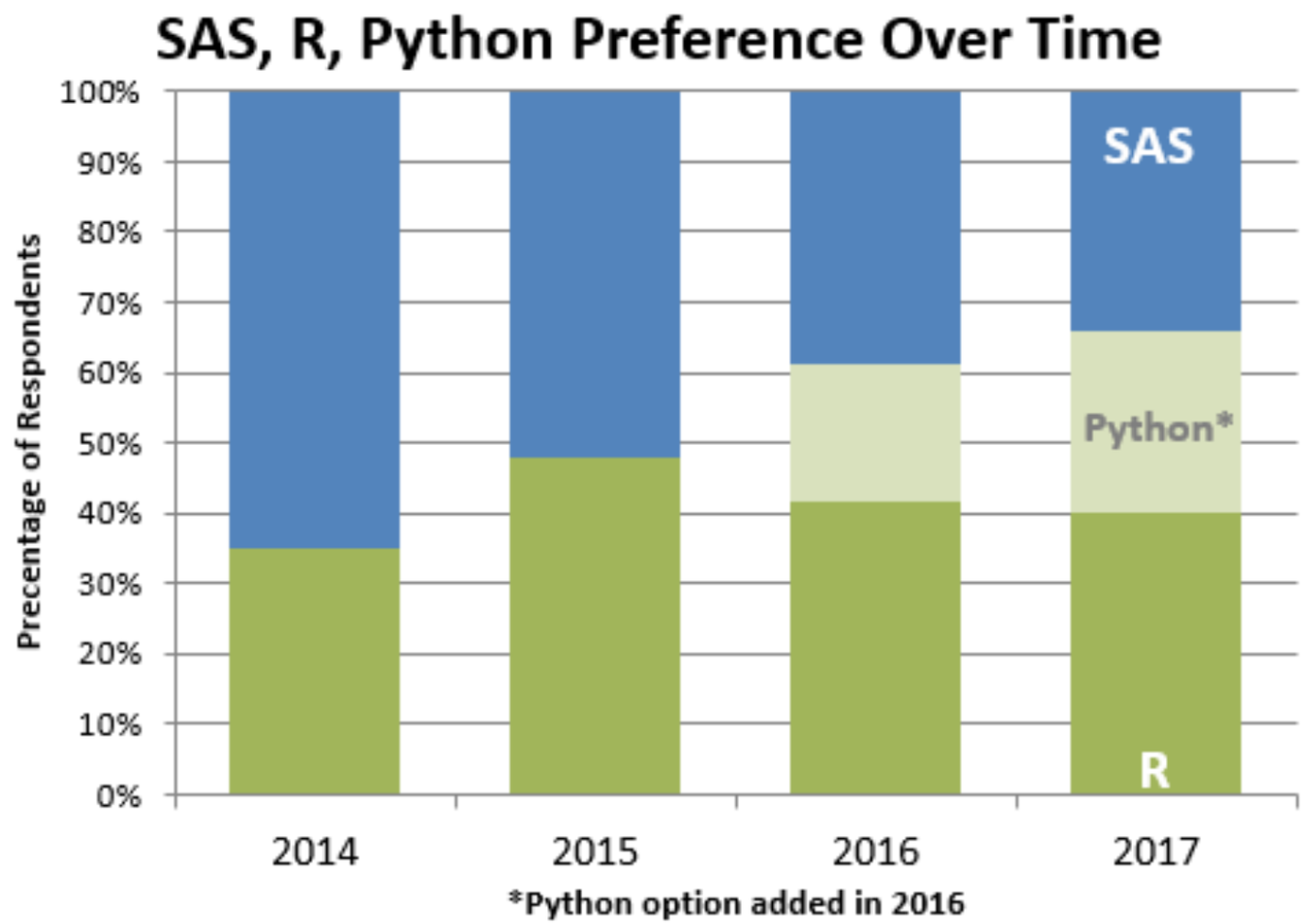


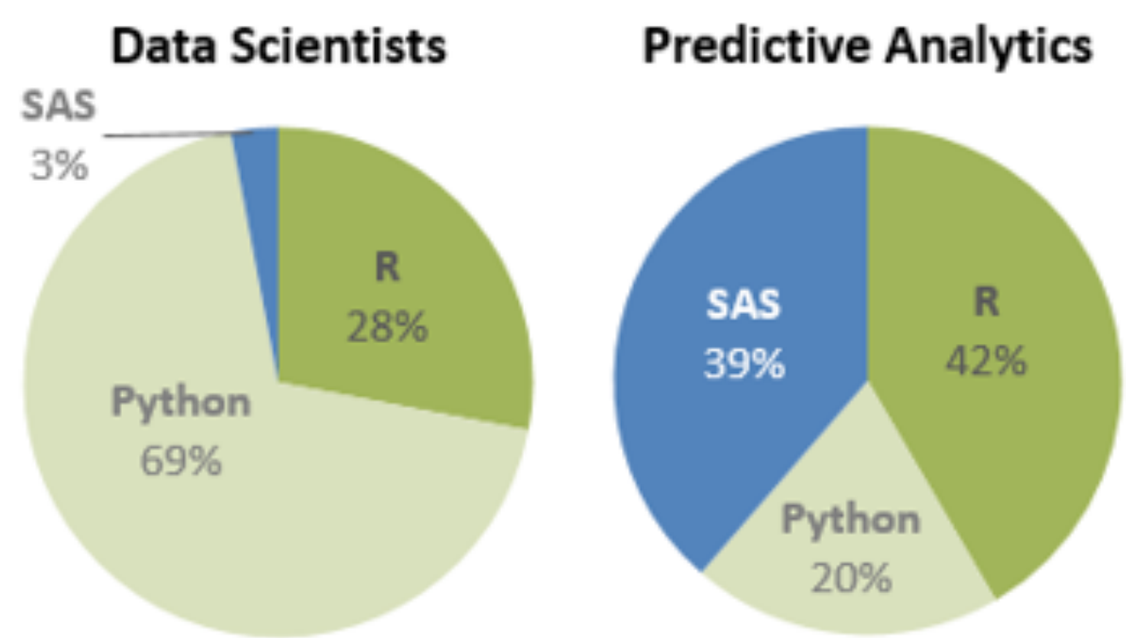


Source: https://www.burtchworks.com/2017/06/19/2017-sas-r-python-flash-survey-results/

R

- R is still widely used, especially in academia
- R is very well suited to do statistical / machine learning
- Due to details of implementation, calculations in R are very efficient





Source: https://www.burtchworks.com/2017/06/19/2017-sas-r-python-flash-survey-results/

PL/R

PL/R

- Procedural language that allows to write PostgreSQL functions and aggregate functions in R

PL/R

- Procedural language that allows to write PostgreSQL functions and aggregate functions in R
- Developed by Joe Conway since 2003

PL/R

- Procedural language that allows to write PostgreSQL functions and aggregate functions in R
- Developed by Joe Conway since 2003
- Implements full R functionality

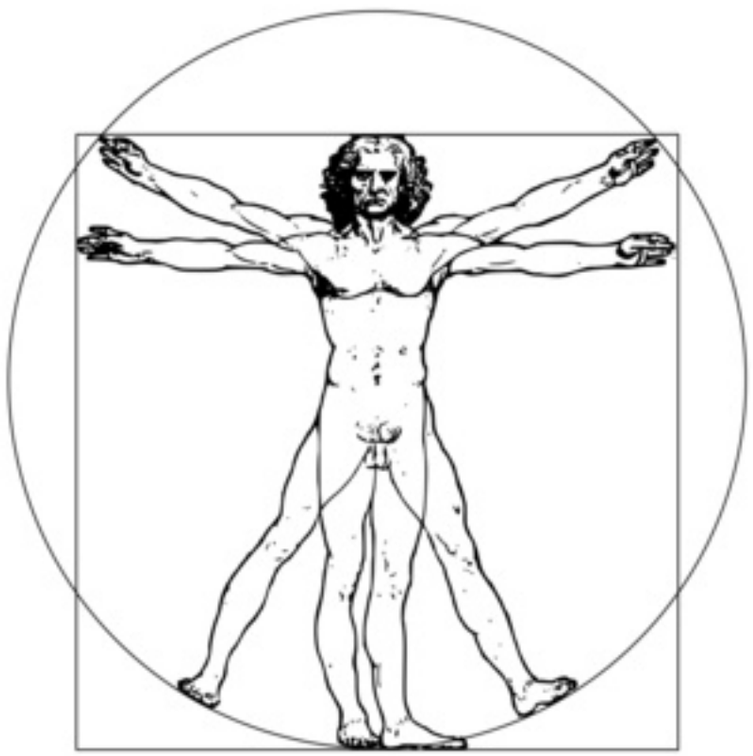
This talk

- No technical details of implementation or management
- User perspective

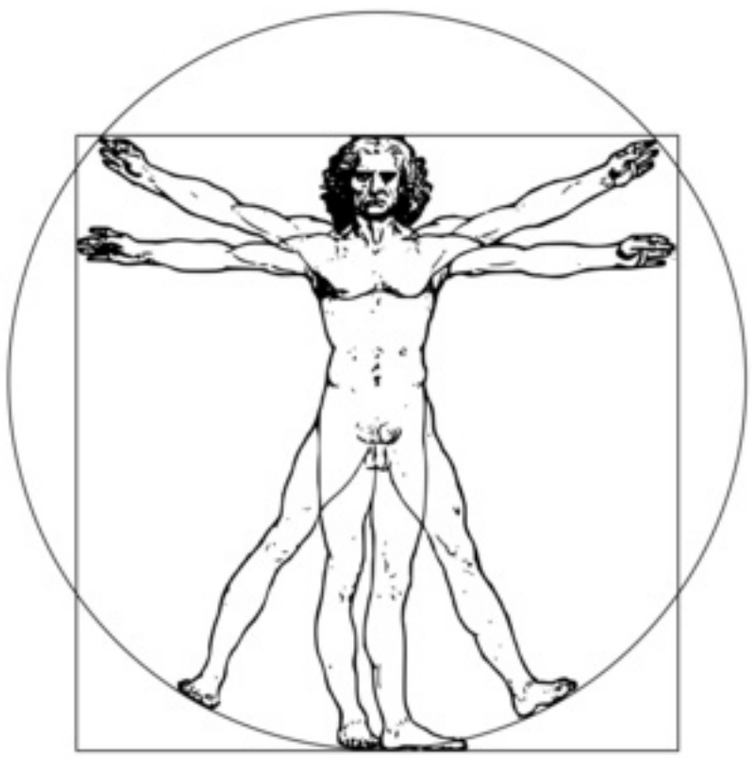
Is it possible to do full cycle of data analysis using only PL/R?

Biology for dummies

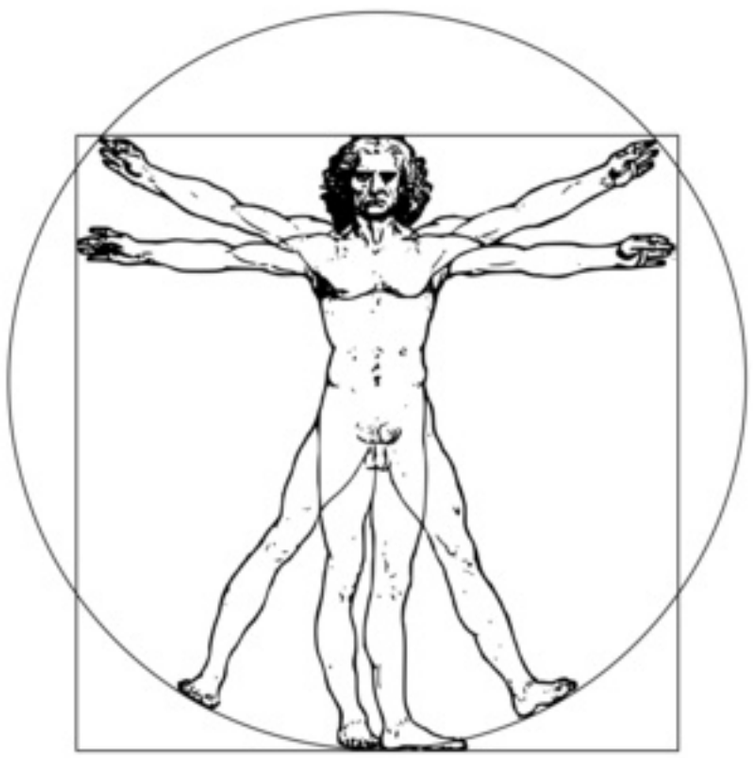
Biology for dummies

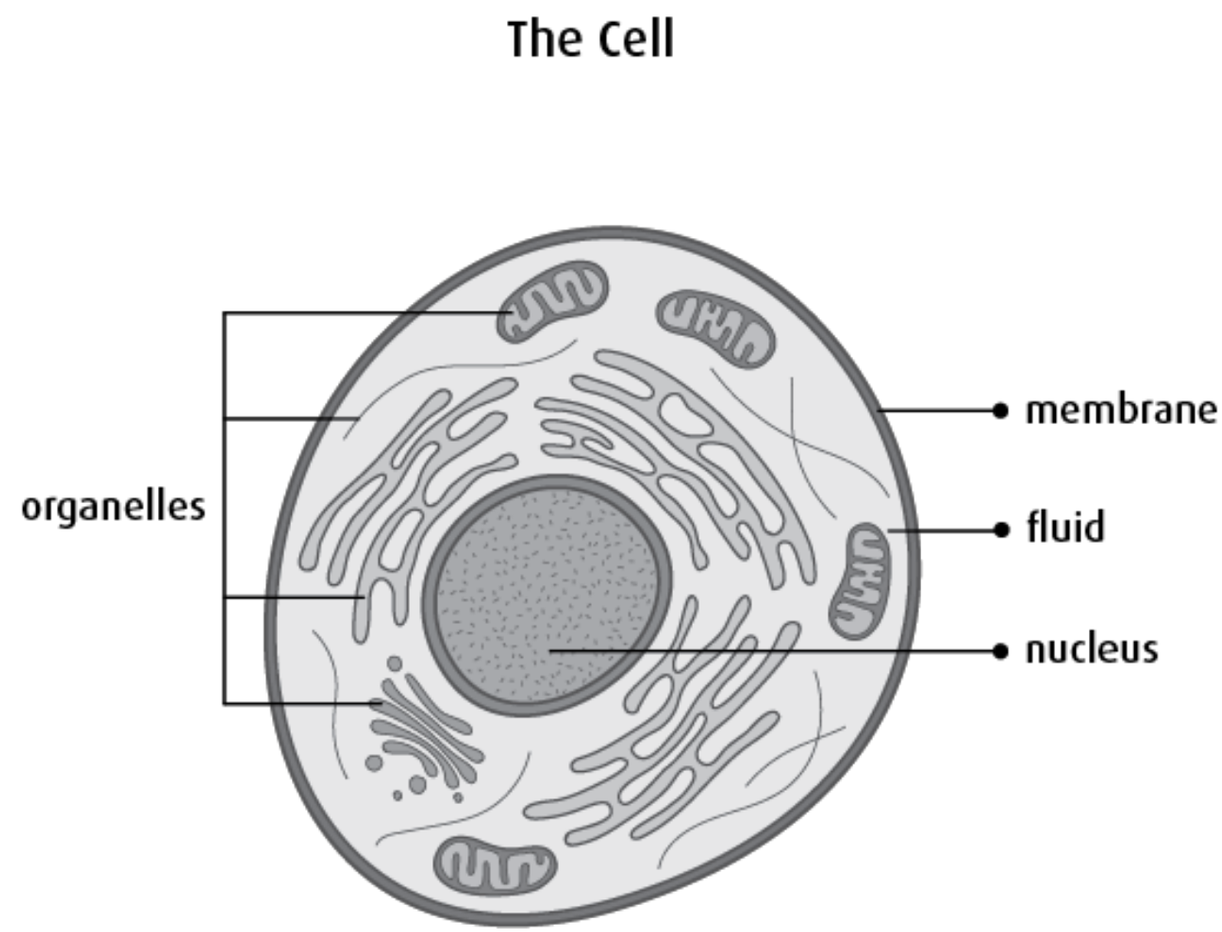


Biology for dummies

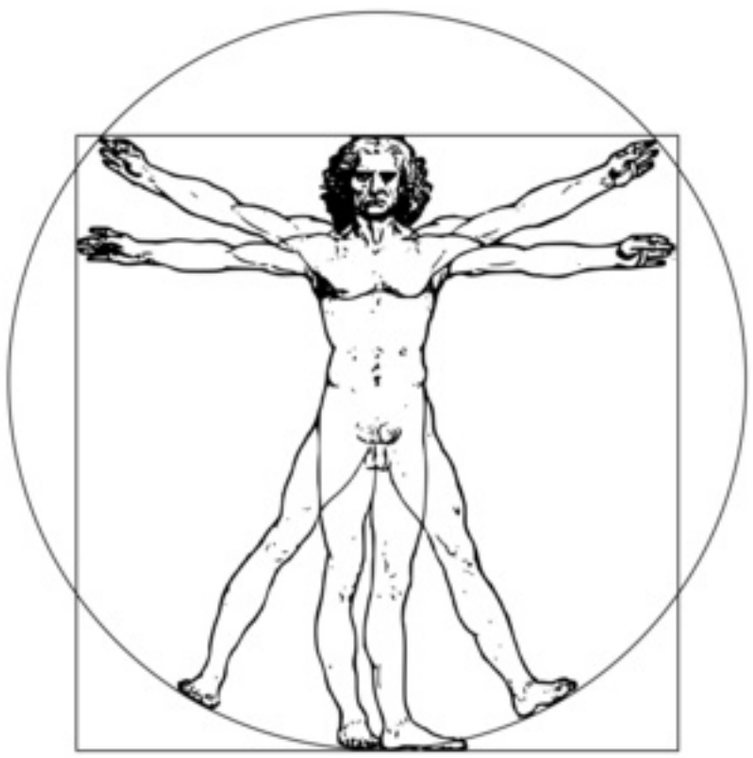


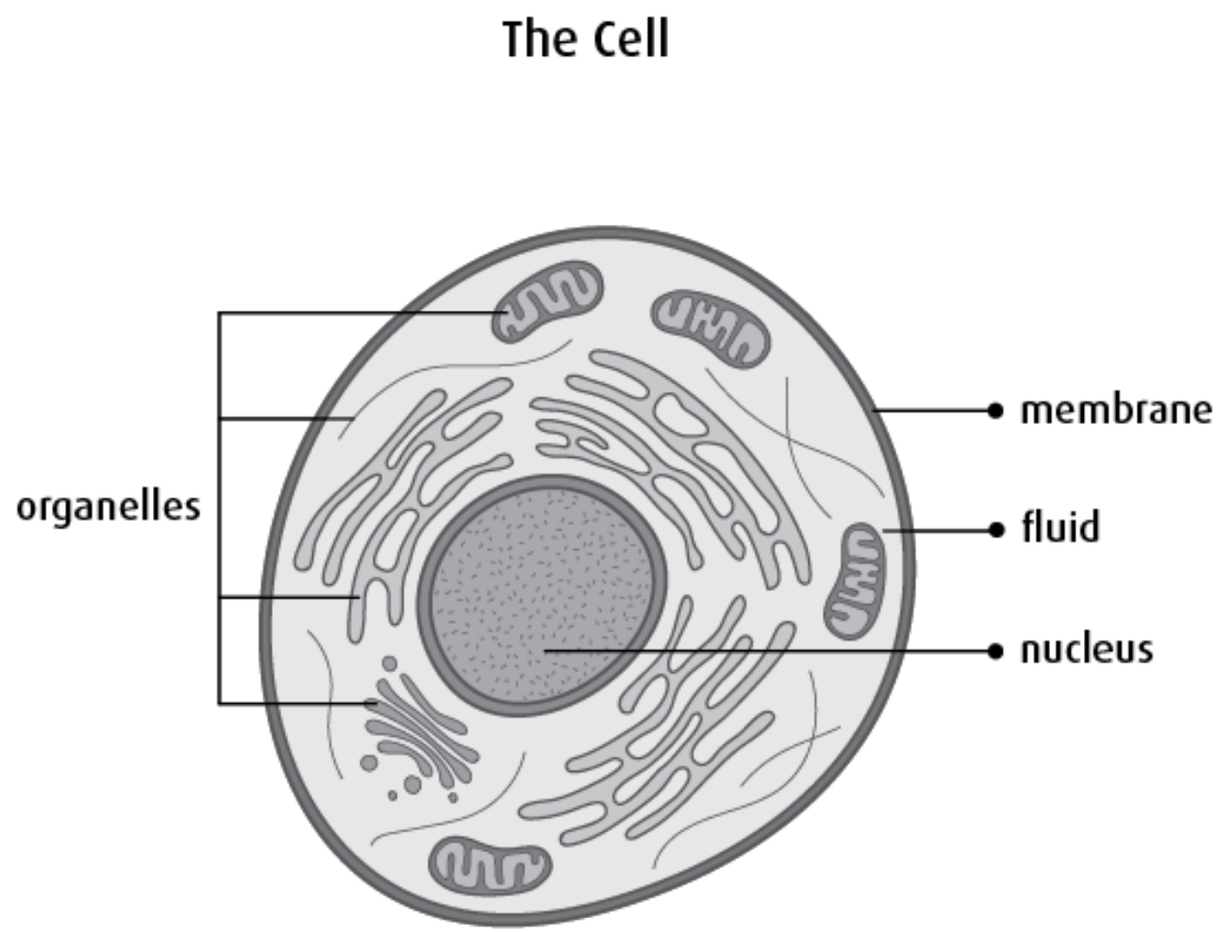
Biology for dummies



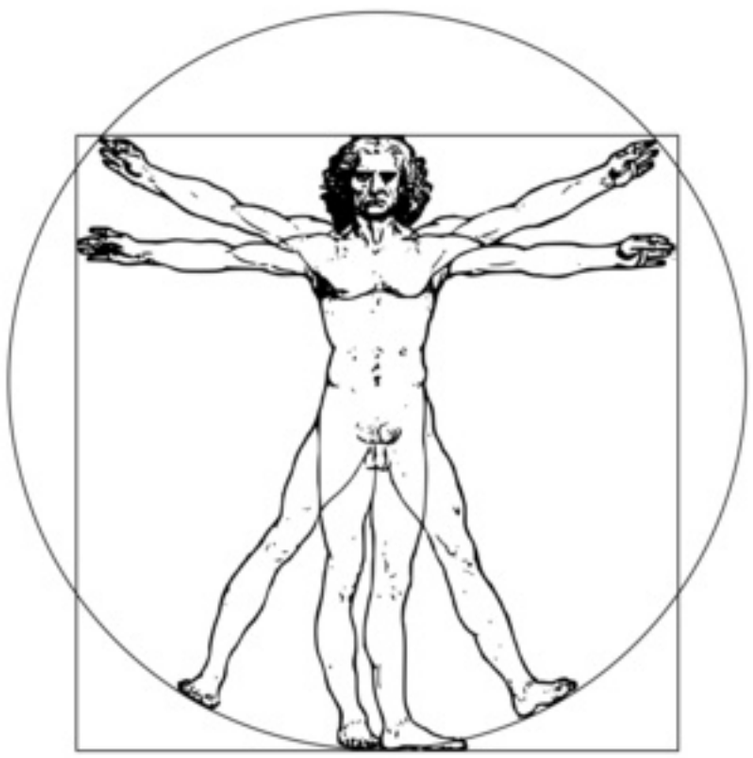


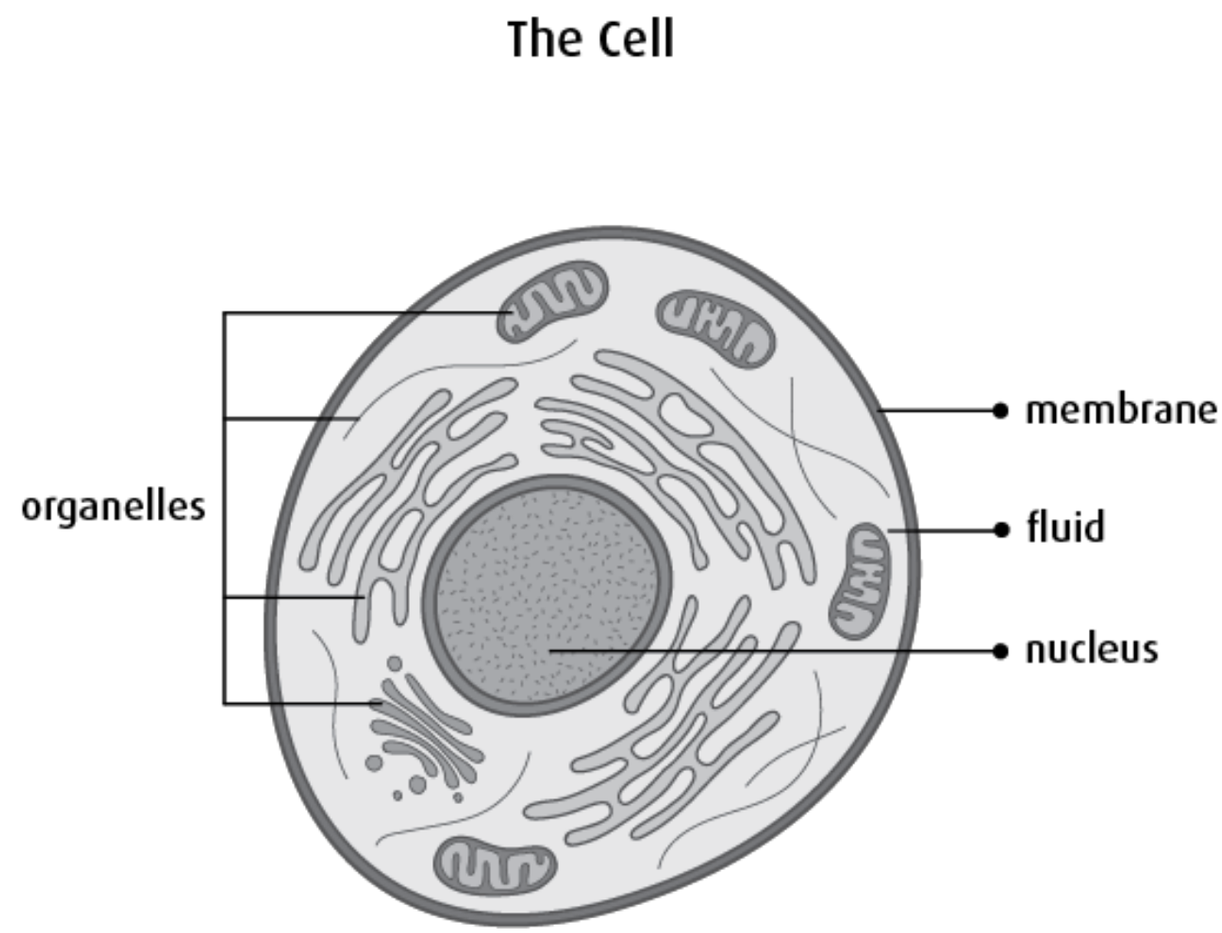
Biology for dummies





Biology for dummies





Biological molecules: DNA, RNA, proteins

Proteins

Proteins

- Biological machines, responsible for (almost) all processes within the cell

Proteins

- Biological machines, responsible for (almost) all processes within the cell

- Encoded in genome as a sequence of characters

Proteins

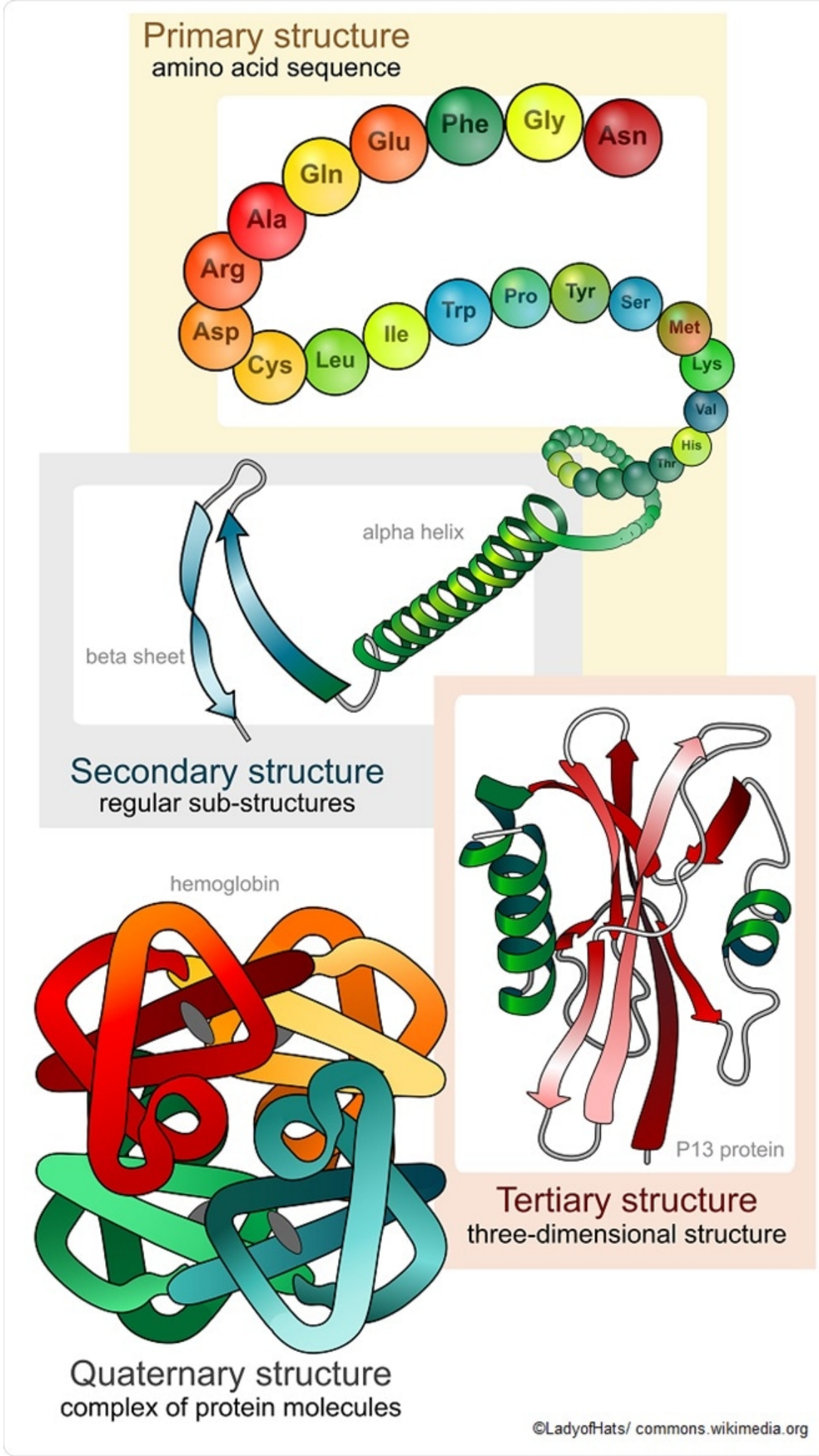
- Biological machines, responsible for (almost) all processes within the cell
- Encoded in genome as a sequence of characters
- => synthesized as a chain of similar, yet not identical (chemically) units

Proteins

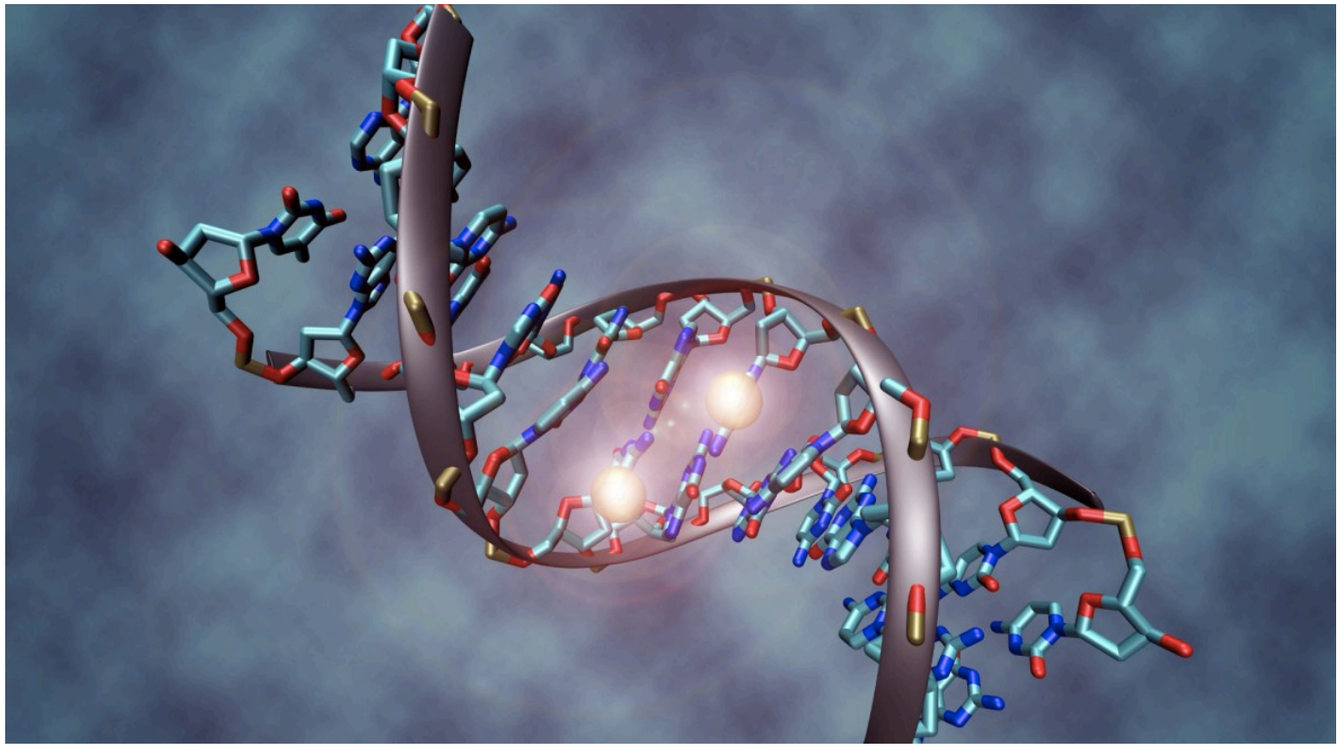
- Biological machines, responsible for (almost) all processes within the cell
- Encoded in genome as a sequence of characters
- => synthesized as a chain of similar, yet not identical (chemically) units
- Folded into 3D structures that makes them functional

Proteins

- Biological machines, responsible for (almost) all processes within the cell
- Encoded in genome as a sequence of characters
- => synthesized as a chain of similar, yet not identical (chemically) units
- Folded into 3D structures that makes them functional

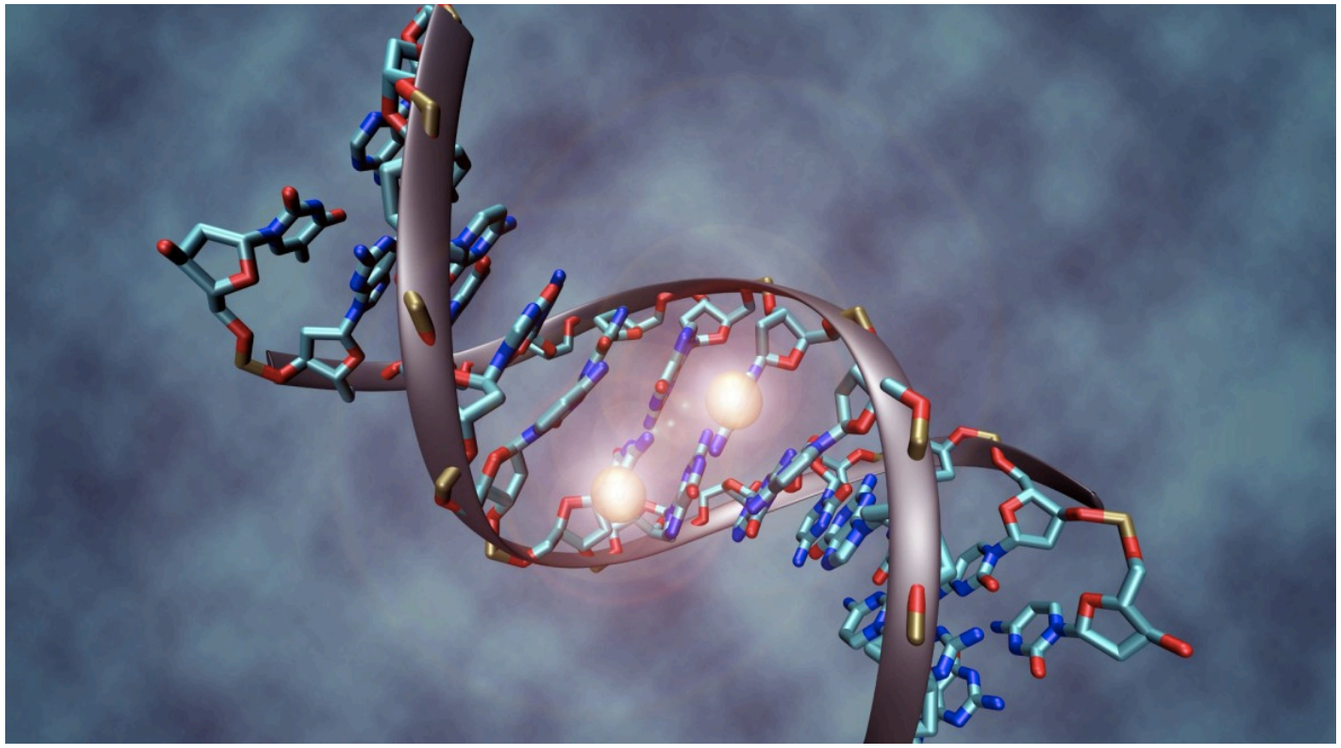


Mutations

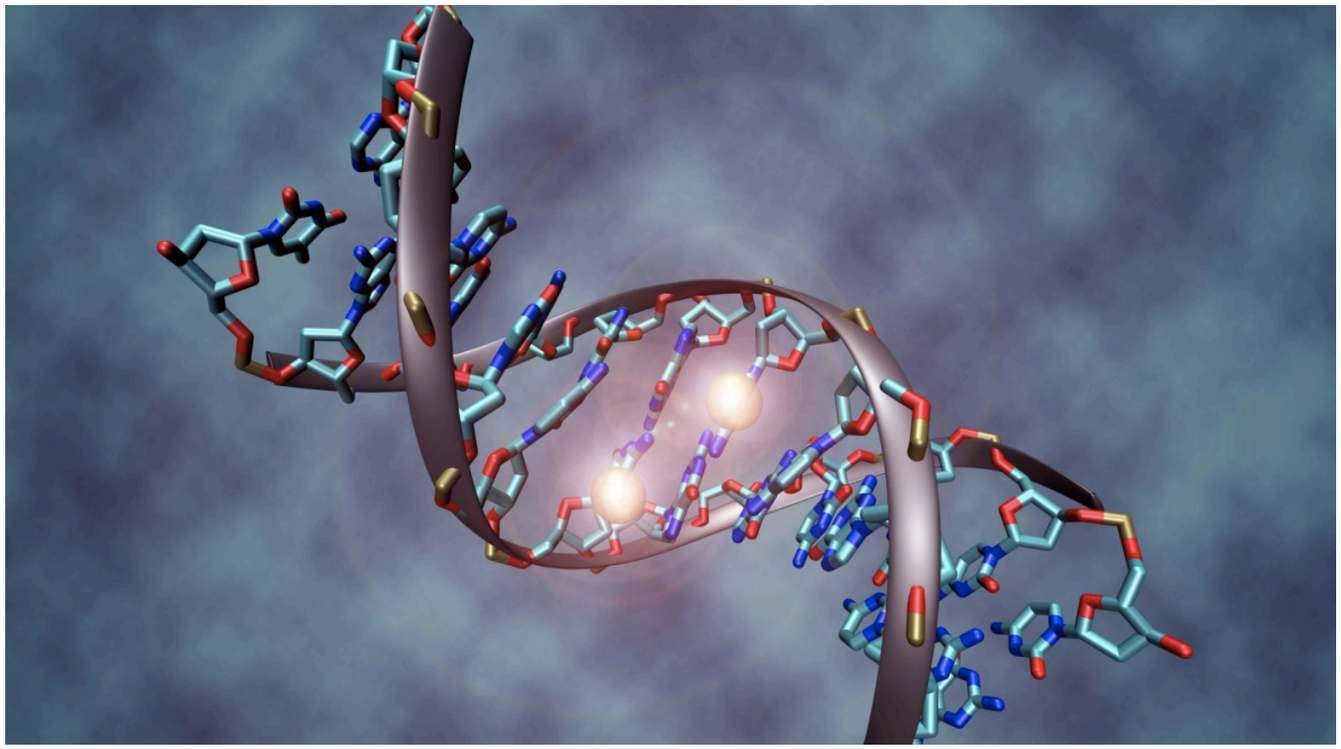


Mutations

- Happen in DNA

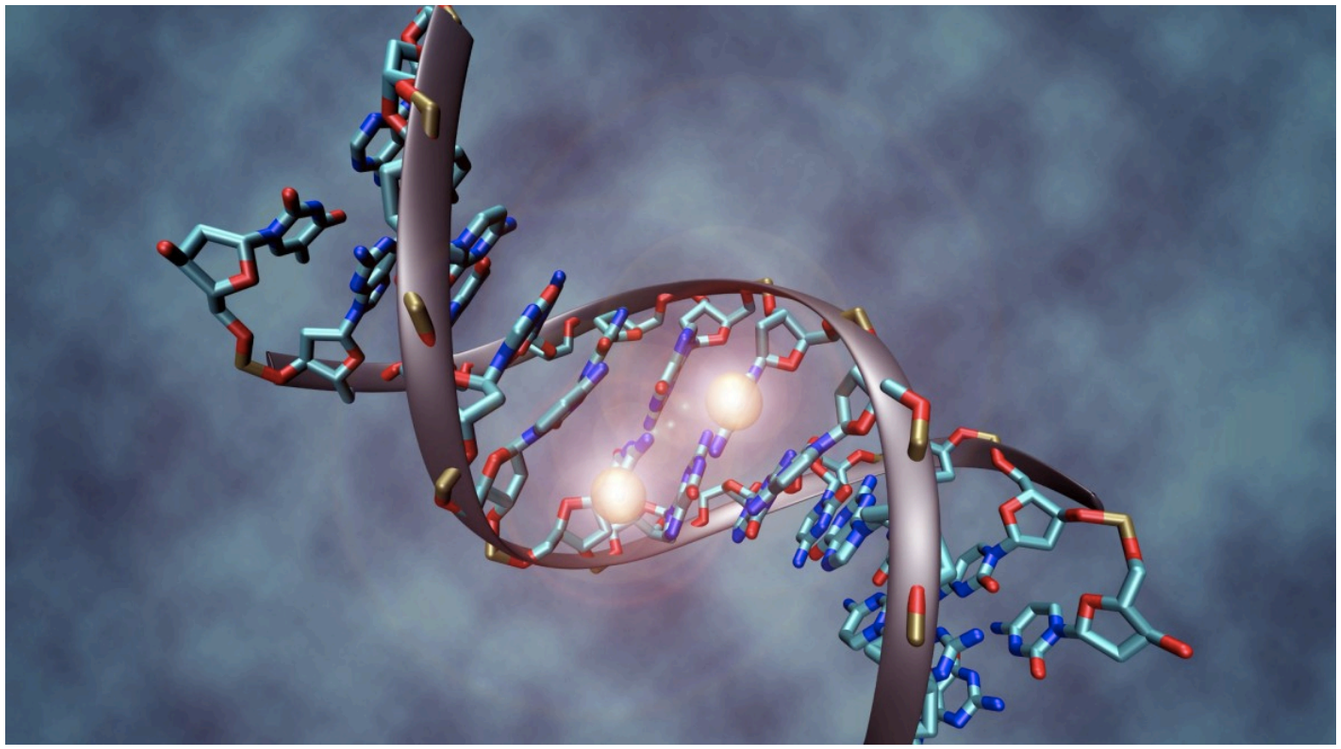


Mutations



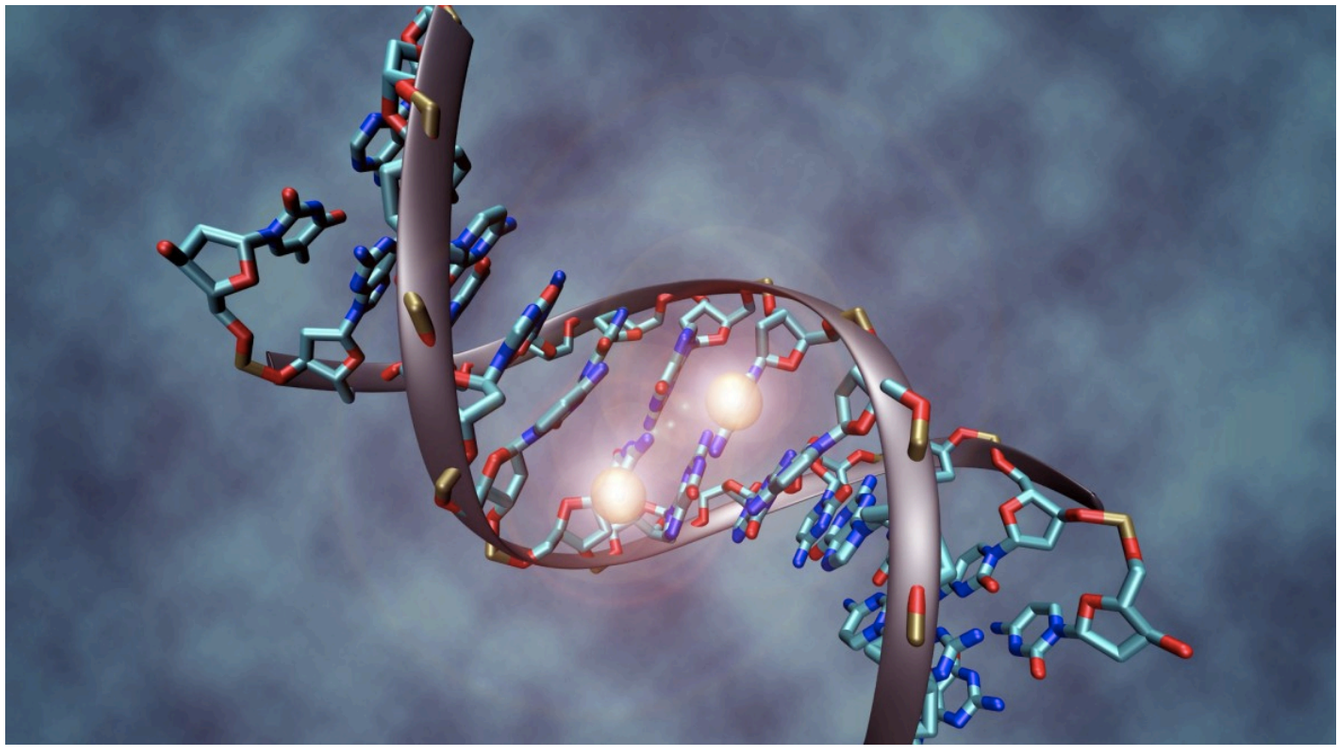
- Happen in DNA
- Sources:
  - Spontaneous mistakes of DNA polymerase
  - Endogenous DNA damage
  - Exogenous DNA damage

Mutations



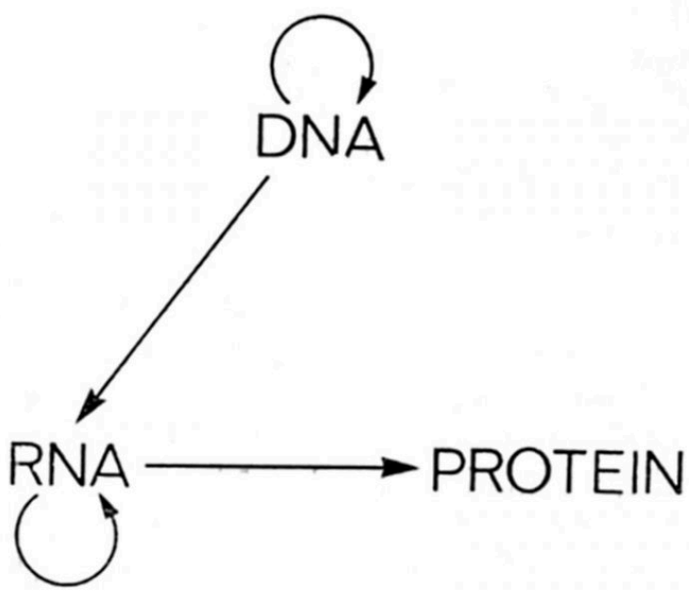
- Happen in DNA
- Sources:
  - Spontaneous mistakes of DNA polymerase
  - Endogenous DNA damage
  - Exogenous DNA damage
- Repair mechanisms => 1 mutation in **10^{10}** nucleotides per cell division

Mutations



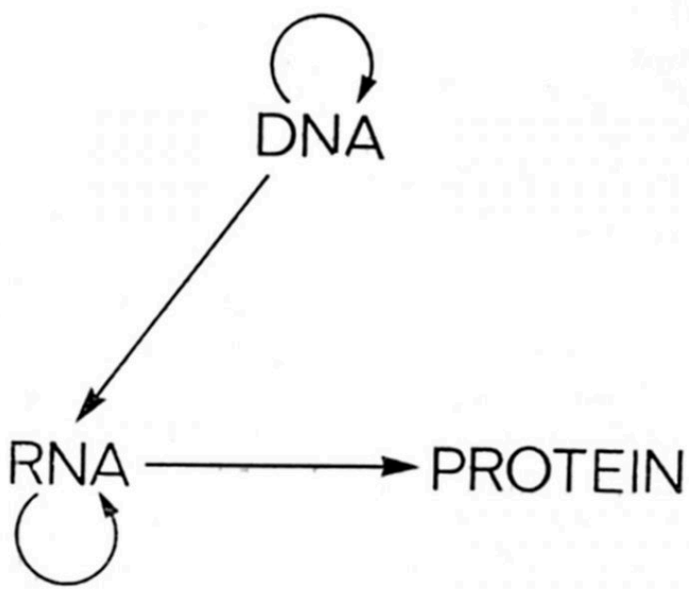
- Happen in DNA
- Sources:
  - Spontaneous mistakes of DNA polymerase
  - Endogenous DNA damage
  - Exogenous DNA damage
- Repair mechanisms => 1 mutation in **10^{10}** nucleotides per cell division
- Cf. human genome size: **3×10^{9}** bp

The Central Dogma: flow of information in the living cells

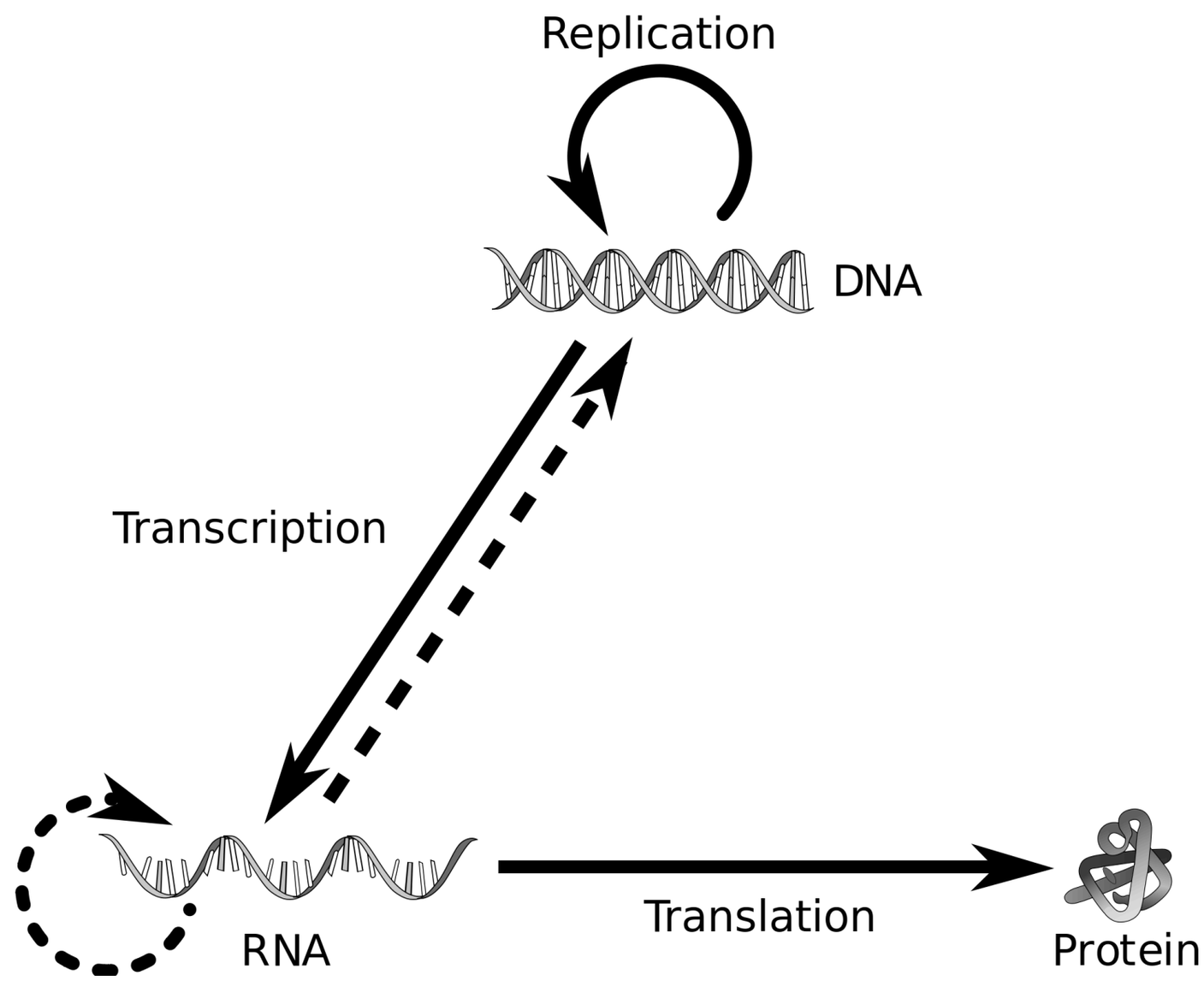


Crick's 1958 version of the "central dogma" of biology from Crick (1970, Figure 3, p. 562).

The Central Dogma: flow of information in the living cells

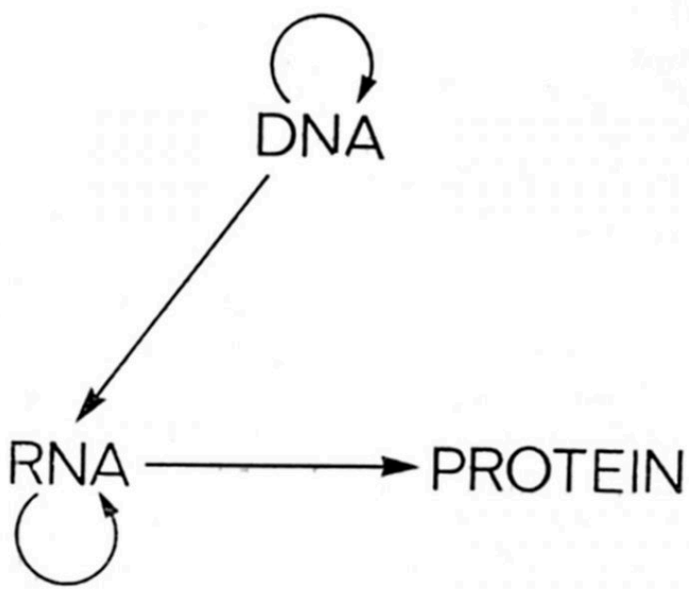


Crick's 1958 version of the "central dogma" of biology from Crick (1970, Figure 3, p. 562).

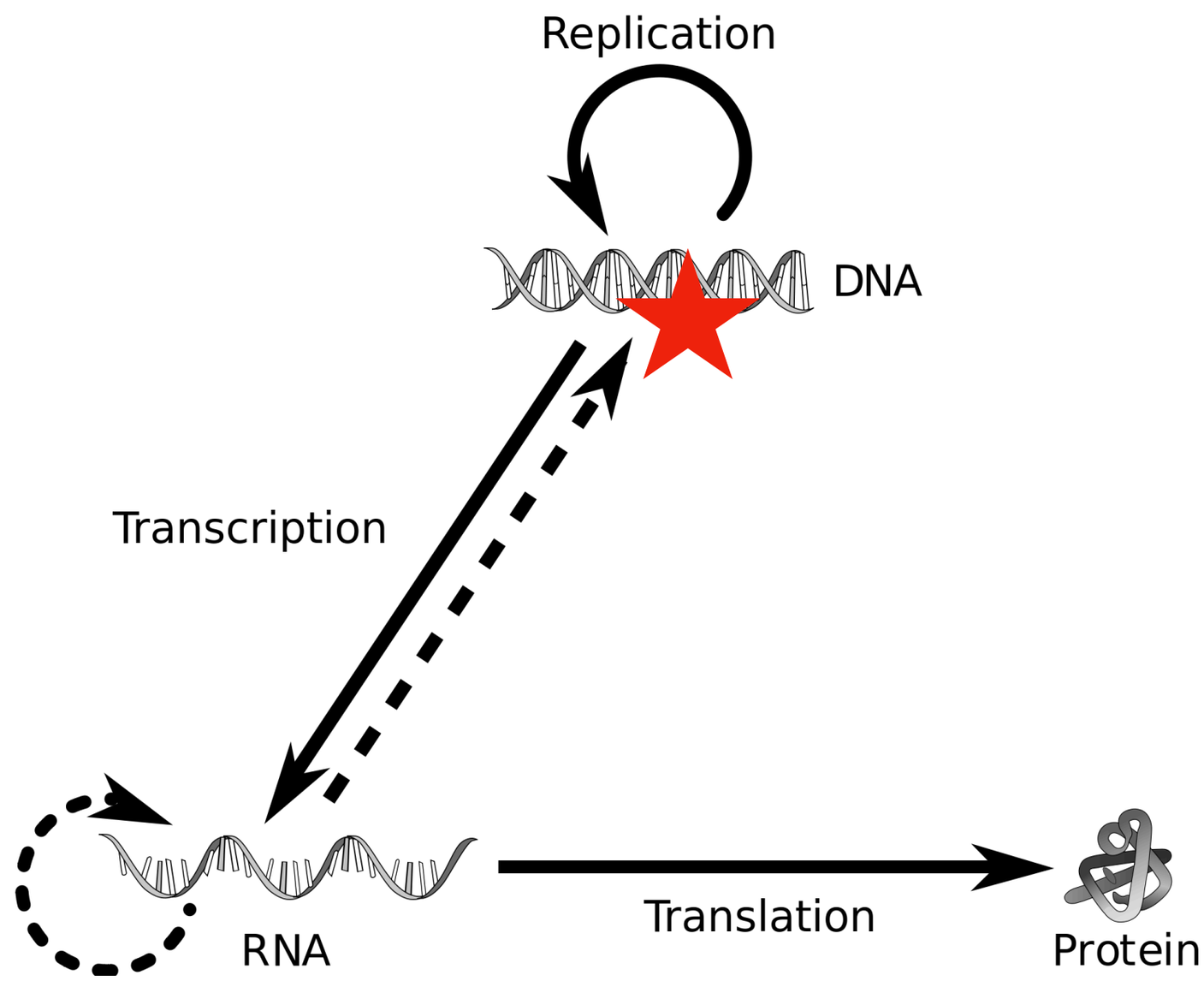


https://commons.wikimedia.org/wiki/File:Central_dogma_of_molecular_biology.svg

The Central Dogma: flow of information in the living cells

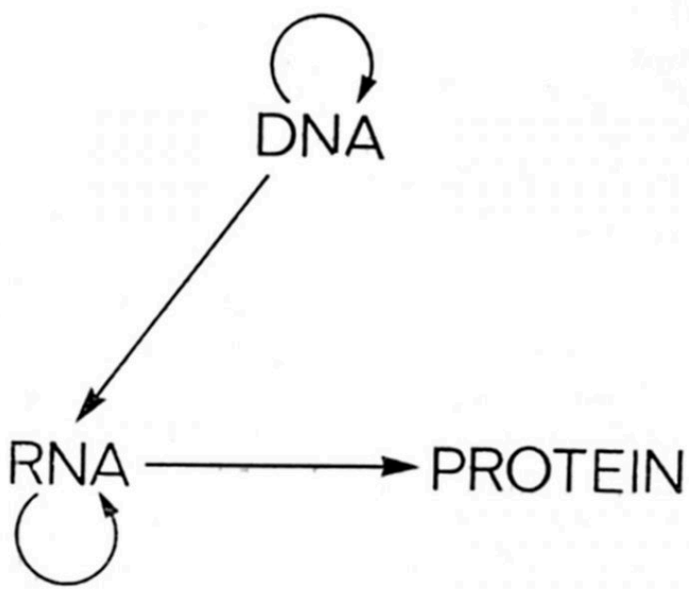


Crick's 1958 version of the "central dogma" of biology from Crick (1970, Figure 3, p. 562).

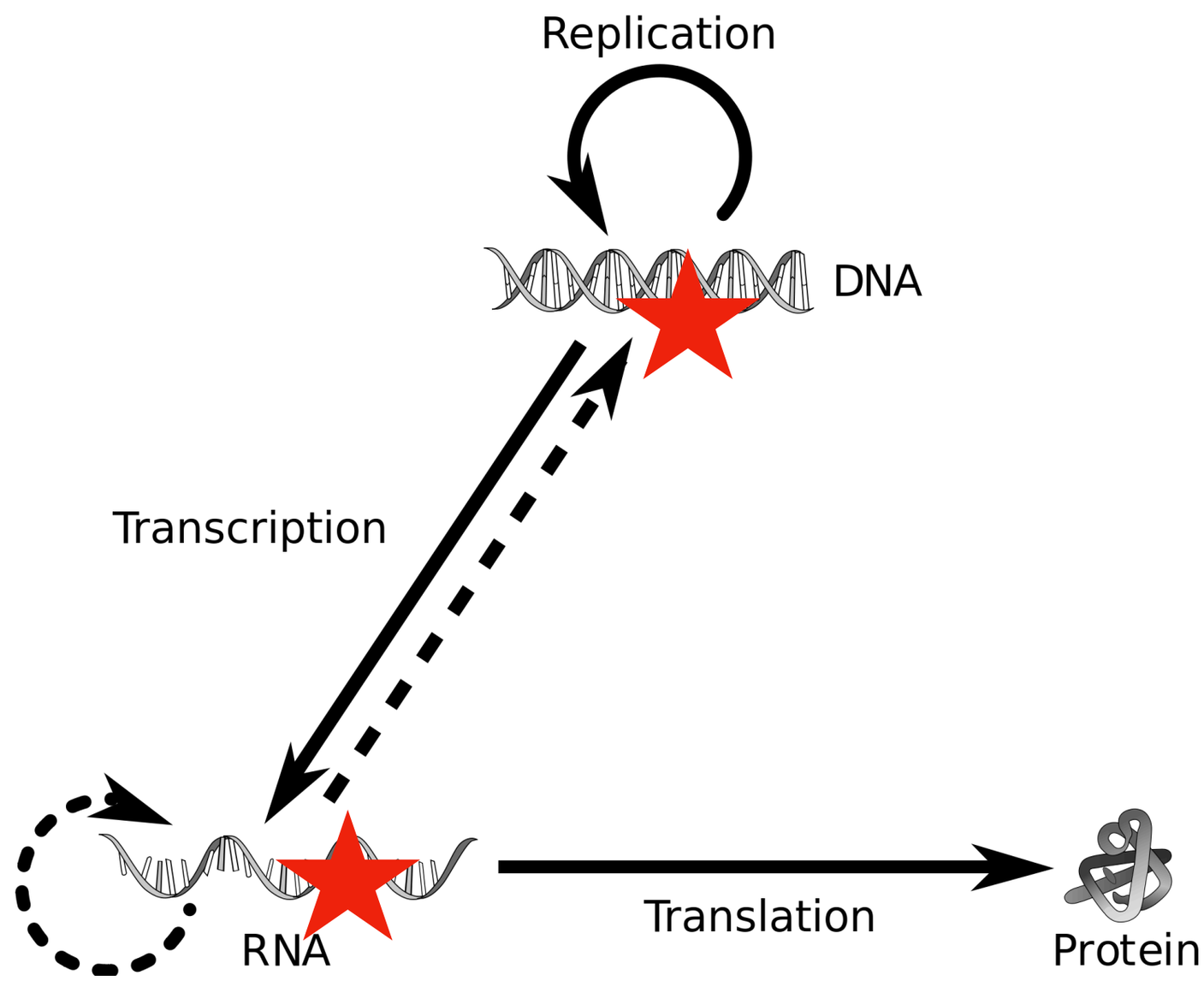


https://commons.wikimedia.org/wiki/File:Central_dogma_of_molecular_biology.svg

The Central Dogma: flow of information in the living cells

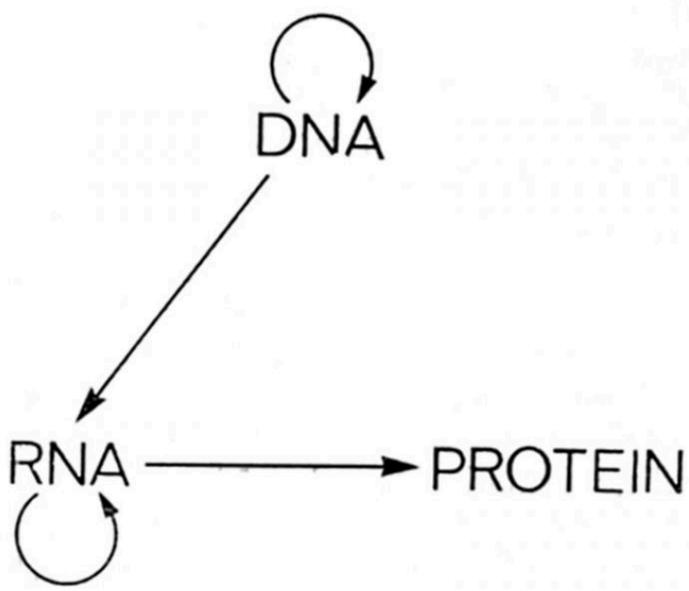


Crick's 1958 version of the "central dogma" of biology from Crick (1970, Figure 3, p. 562).

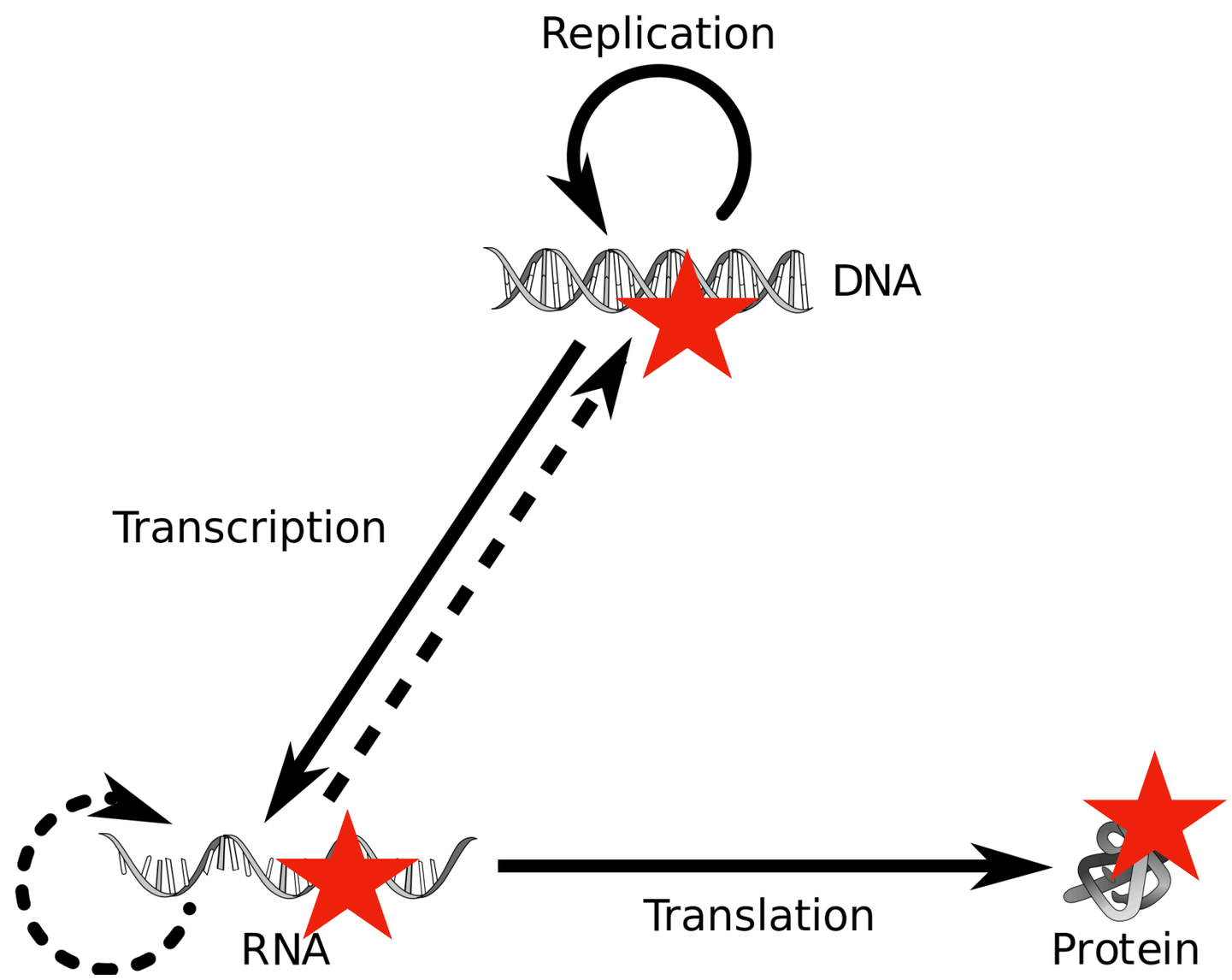


https://commons.wikimedia.org/wiki/File:Central_dogma_of_molecular_biology.svg

The Central Dogma: flow of information in the living cells



Crick's 1958 version of the "central dogma" of biology from Crick (1970, Figure 3, p. 562).



https://commons.wikimedia.org/wiki/File:Central_dogma_of_molecular_biology.svg

Protein thermodynamic stability

Protein thermodynamic stability

- Simple case: protein can unfold and refold rapidly, reversibly, via a two-state mechanism

Protein thermodynamic stability

- Simple case: protein can unfold and refold rapidly, reversibly, via a two-state mechanism

- $\Delta G = G_{\text{unfolded}} - G_{\text{folded}}$

Protein thermodynamic stability

- Simple case: protein can unfold and refold rapidly, reversibly, via a two-state mechanism

- $\Delta G = G_{\text{unfolded}} - G_{\text{folded}}$

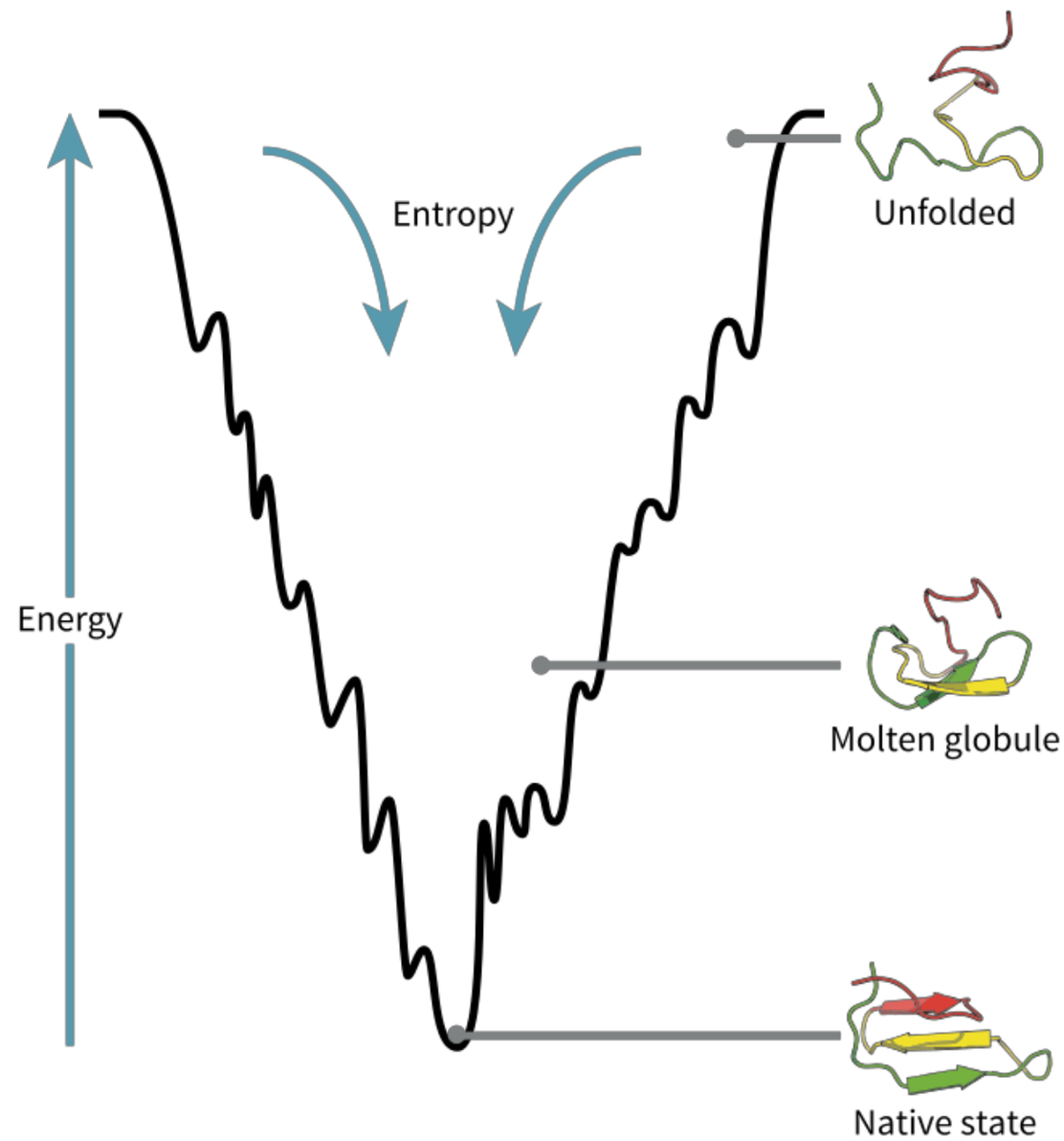
- Upon mutations, ΔG can change:
  $\Delta\Delta G = \Delta G^{\text{mut}} - \Delta G^{\text{WT}}$

Protein thermodynamic stability

- Simple case: protein can unfold and refold rapidly, reversibly, via a two-state mechanism

- $\Delta G = G_{\text{unfolded}} - G_{\text{folded}}$

- Upon mutations, ΔG can change:
  $\Delta\Delta G = \Delta G^{\text{mut}} - \Delta G^{\text{WT}}$



https://commons.wikimedia.org/w/index.php?curid=28353539

Some data (real-life)

- **ΔΔG** estimates upon mutations

#chr	Gene	ClinicalSignificance	uniprot_ac	uniprot_pos	aa1	aa2	FX_ddG
chr1	ISG15	Benign	P05161	83	S	N	-0.517133
chr2	DNMT3A	Pathogenic	Q9Y6K1	583	C	Y	33.0787
chr1	AGRN	Benign	O00468-6	15	P	R	?

…

- 84,426 rows (13 MB)

Reading the data (R)

```
> x<-read.table("clinvar.main.pph.ddg.uniprot.tsv",
sep='\t', header=T)
> x[ x == "?" ] <- NA
> nrow(x)
84426
```

- => **data frame**

Reading the data (Postgres)

```
kalinina=# CREATE TABLE clinvar (chr text, to1 bigint, ref text,
alt text, GeneSymbol text, ClinicalSignificance text,
ReviewStatus text, PhenotypeList text, uniprot_ac text,
uniprot_pos int, aa1 char(1), aa2 char(1), prediction text,
PDB_id text, PDB_pos text, PDB_ch char(1), ident float, FX_ddG
float, IM_ddG float, M_ddG float, M_conf float);
CREATE TABLE

kalinina=# COPY clinvar FROM 'clinvar.main.pph.ddg.uniprot.tsv'
WITH (NULL '?', DELIMITER E'\t');
COPY 84426
```

Calculate median (R)

```
>median(x$FX_ddG)
[1] NA
```

Calculate median (R)

```
>median(x$FX_ddG)
[1] NA

>median(x$FX_ddG, na.rm=TRUE)
[1] 0.974858
```

Calculate median (R)

```
>median(x$FX_ddG)
[1] NA

>median(x$FX_ddG, na.rm=TRUE)
[1] 0.974858

>(x[x$ClinicalSignificance=='Pathogenic',]$FX_ddG)
[1] 1.7756
```

Calculate median (R)

```
>median(x$FX_ddG)
[1] NA

>median(x$FX_ddG, na.rm=TRUE)
[1] 0.974858

>(x[x$ClinicalSignificance=='Pathogenic',]$FX_ddG)
[1] 1.7756

> aggregate(FX_ddG ~ ClinicalSignificance, data = x, FUN =
median)
    ClinicalSignificance FX_ddG
1                 Benign 0.62209
2             Pathogenic 1.77560
```

Calculate median (PL/R)

```
kalinina=# CREATE or REPLACE FUNCTION r_median(_float8) RETURNS
float AS '
median(arg1)
' LANGUAGE 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE median (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_median
);
CREATE AGGREGATE

kalinina=# SELECT clinicalsignificance, median(fx_ddg) FROM clinvar
GROUP BY clinicalsignificance ORDER BY clinicalsignificance;

clinicalsignificance | median
---------------------+----------
Benign               | 0.6220875
Pathogenic           | 1.7756
(2 rows)
```

Summary statistics (R)

```
> aggregate(FX_ddG ~ ClinicalSignificance, data = x, FUN = summary)
    ClinicalSignificance FX_ddG.Min. FX_ddG.1st Qu. FX_ddG.Median FX_ddG.Mean FX_ddG.3rd Qu. FX_ddG.Max.
1                 Benign    -5.77969       -0.04082       0.62209     1.37172       1.91954     62.08970
2             Pathogenic   -18.09830        0.30438       1.77560     3.21887       4.21793     52.26050
```

Summary statistics (R)

```
> aggregate(FX_ddG ~ ClinicalSignificance, data = x, FUN = summary)
    ClinicalSignificance FX_ddG.Min. FX_ddG.1st Qu. FX_ddG.Median FX_ddG.Mean FX_ddG.3rd Qu. FX_ddG.Max.
1                 Benign    -5.77969       -0.04082       0.62209     1.37172       1.91954     62.08970
2             Pathogenic   -18.09830        0.30438       1.77560     3.21887       4.21793     52.26050
```


```
> aggregate(FX_ddG ~ ClinicalSignificance, data = x, FUN = summary)
    ClinicalSignificance FX_ddG.Min. FX_ddG.1st Qu. FX_ddG.Median
1                 Benign    -5.77969       -0.04082       0.62209
2             Pathogenic   -18.09830        0.30438       1.77560

FX_ddG.Mean FX_ddG.3rd Qu. FX_ddG.Max.
    1.37172       1.91954     62.08970
    3.21887       4.21793     52.26050
```

Summary statistics (R)

```
> aggregate(FX_ddG ~ ClinicalSignificance, data = x, FUN = summary)
    ClinicalSignificance FX_ddG.Min. FX_ddG.1st Qu. FX_ddG.Median FX_ddG.Mean FX_ddG.3rd Qu. FX_ddG.Max.
1                 Benign    -5.77969       -0.04082       0.62209     1.37172       1.91954     62.08970
2             Pathogenic   -18.09830        0.30438       1.77560     3.21887       4.21793     52.26050
```

```
> aggregate(FX_ddG ~ ClinicalSignificance, data = x, FUN = summary)
    ClinicalSignificance FX_ddG.Min. FX_ddG.1st Qu. FX_ddG.Median
1                 Benign    -5.77969       -0.04082       0.62209
2             Pathogenic   -18.09830        0.30438       1.77560

FX_ddG.Mean FX_ddG.3rd Qu. FX_ddG.Max.
    1.37172       1.91954     62.08970
    3.21887       4.21793     52.26050
```

You need additional code if you need to preserve a specific order of categories

Summary statistics (PL/R)

```
kalinina=# CREATE or REPLACE FUNCTION r_summary(_float8) RETURNS _float8 AS '
summary(arg1)
' LANGUAGE 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE summary (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_median
);
CREATE AGGREGATE

kalinina=# SELECT clinicalsignificance, SELECT summary(fx_ddg) FROM clinvar GROUP BY
clinicalsignificance ORDER BY clinicalsignificance;

clinicalsignificance | summary
---------------------+-----------------------------------------------------------------
Benign               | {-5.77969,-0.040819875,0.6220875,1.37171750416516,1.9195375,62.0897}
Pathogenic           | {-18.0983,0.3043845,1.7756,3.21886833468419,4.217925,52.2605}
(2 rows)
```

Boxplot (R)

```
>boxplot(x[ x$ClinicalSignificance == ‘Pathogenic’, ]$FX_ddG)
```

Boxplot (R)

```
>boxplot(x[ x$ClinicalSignificance == ‘Pathogenic’, ]$FX_ddG)
```

Boxplot (R)

```
>boxplot(x[ x$ClinicalSignificance == ‘Pathogenic’, ]$FX_ddG)
```

- Syntax for subsetting:

```
x[ x$<someFactor> == ‘<someValue>’, ]
```

Boxplot (R)

```
>boxplot(x[ x$ClinicalSignificance == ‘Pathogenic’, ]$FX_ddG)
```

- Syntax for subsetting:
  `x[ x$<someFactor> == ‘<someValue>’, ]`

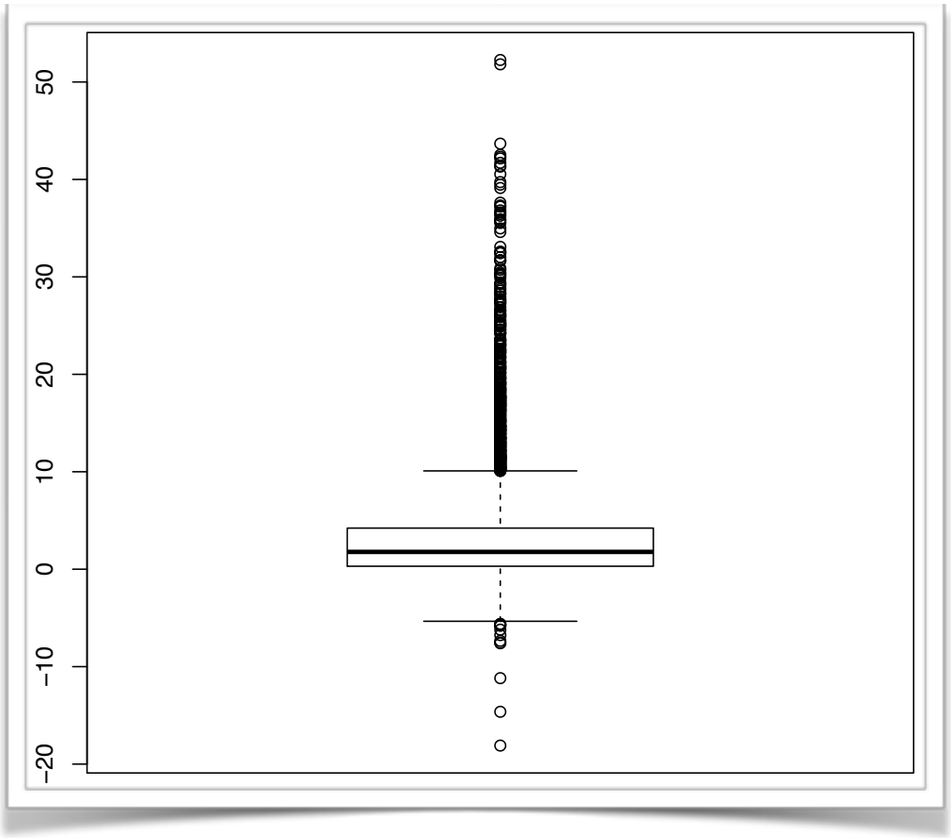
- Output directly to active graphic device

Boxplot (R)

```
>boxplot(x[ x$ClinicalSignificance == ‘Pathogenic’, ]$FX_ddG)
```

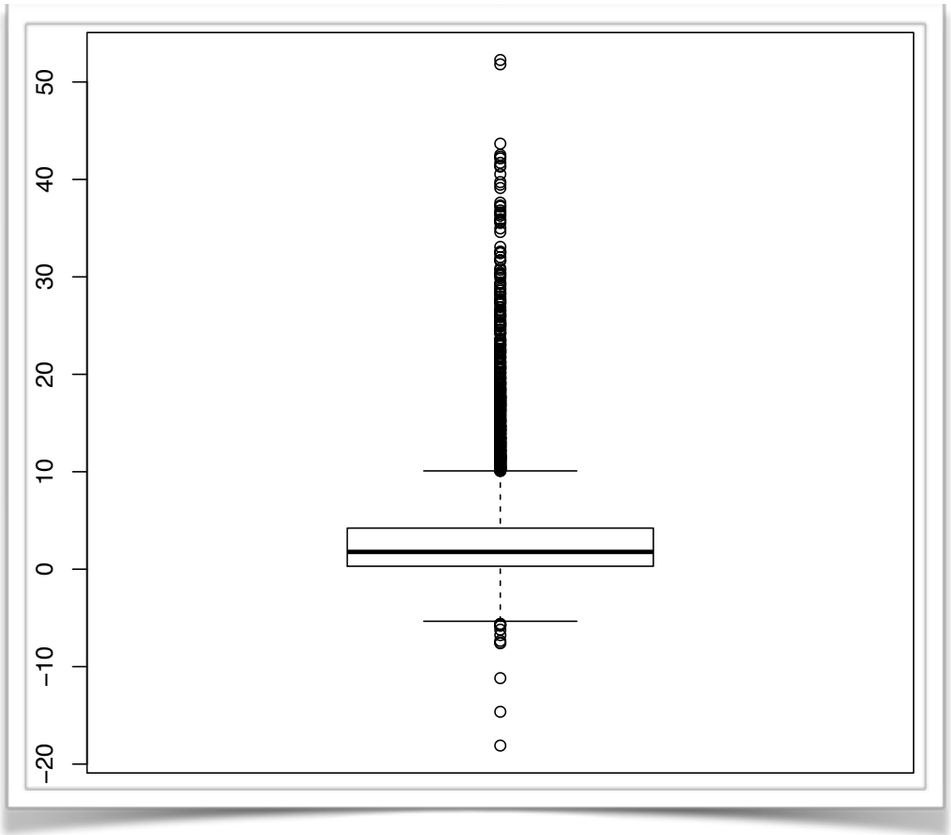
- Syntax for subsetting:
  `x[ x$<someFactor> == ‘<someValue>’, ]`

- Output directly to active graphic device



Boxplot (PL/R)

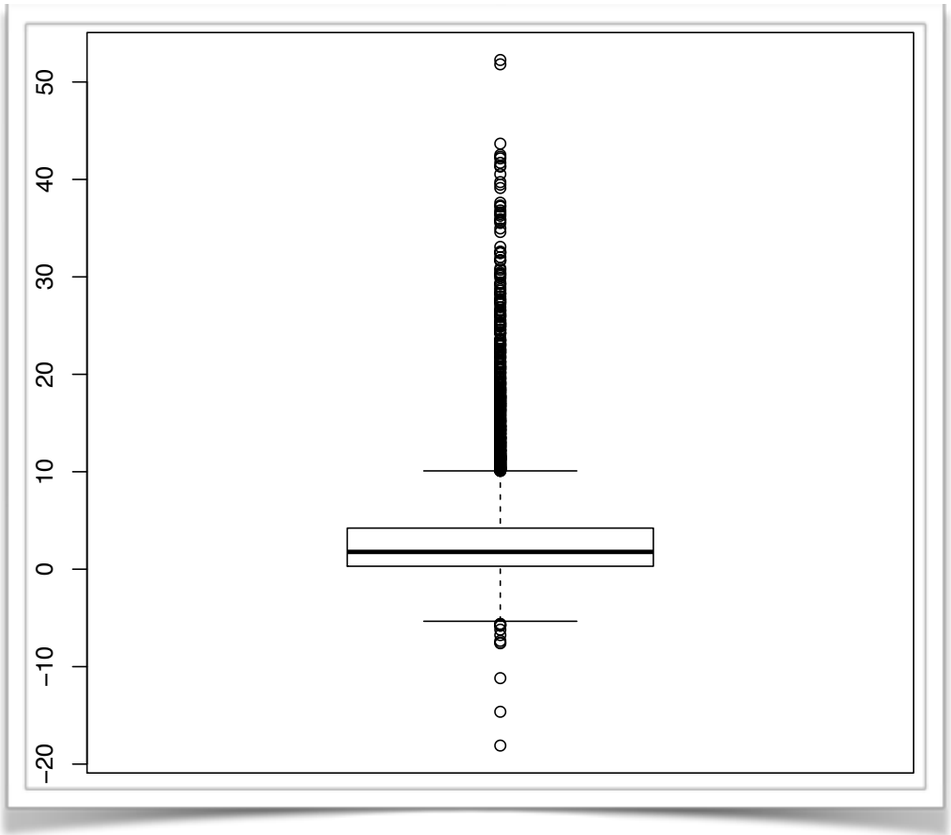
```
CREATE or REPLACE function
r_boxplot2(_float8) RETURNS void AS '
pdf("~/Work/ddG/test.pdf")
boxplot(arg1)
dev.off()
' language 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE boxplot2pdf (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_boxplot2
);
CREATE AGGREGATE

kalinina=# SELECT boxplot2pdf(fx_ddg)
FROM clinvar WHERE clinicalsignificance =
'Pathogenic';
 boxplot2pdf
-------------

(1 row)
```



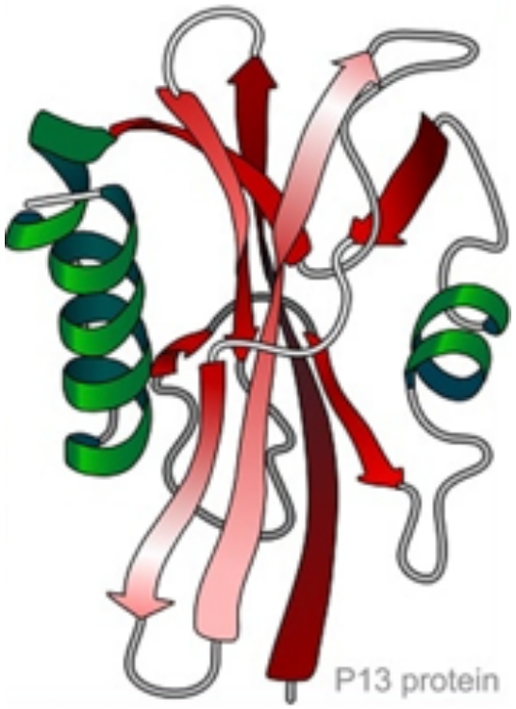
Boxplot (PL/R)

```
CREATE or REPLACE function
r_boxplot2(_float8) RETURNS void AS '
pdf("~/Work/ddG/test.pdf")
boxplot(arg1)
dev.off()
' language 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE boxplot2pdf (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_boxplot2
);
CREATE AGGREGATE

kalinina=# SELECT boxplot2pdf(fx_ddg)
FROM clinvar WHERE clinicalsignificance =
'Pathogenic';
 boxplot2pdf
-------------

(1 row)
```

Only output to file



More data (real-life)



- **Structural annotation** of the human proteome

#AC	Mut	Species	Tags	Surface/Core	**Class**
P30613	R498	HUMAN	None	Surface	**Ligand**
P30613	G411	HUMAN	None	Core	**Core**
P30613	R559	HUMAN	None	None	**Disorder**

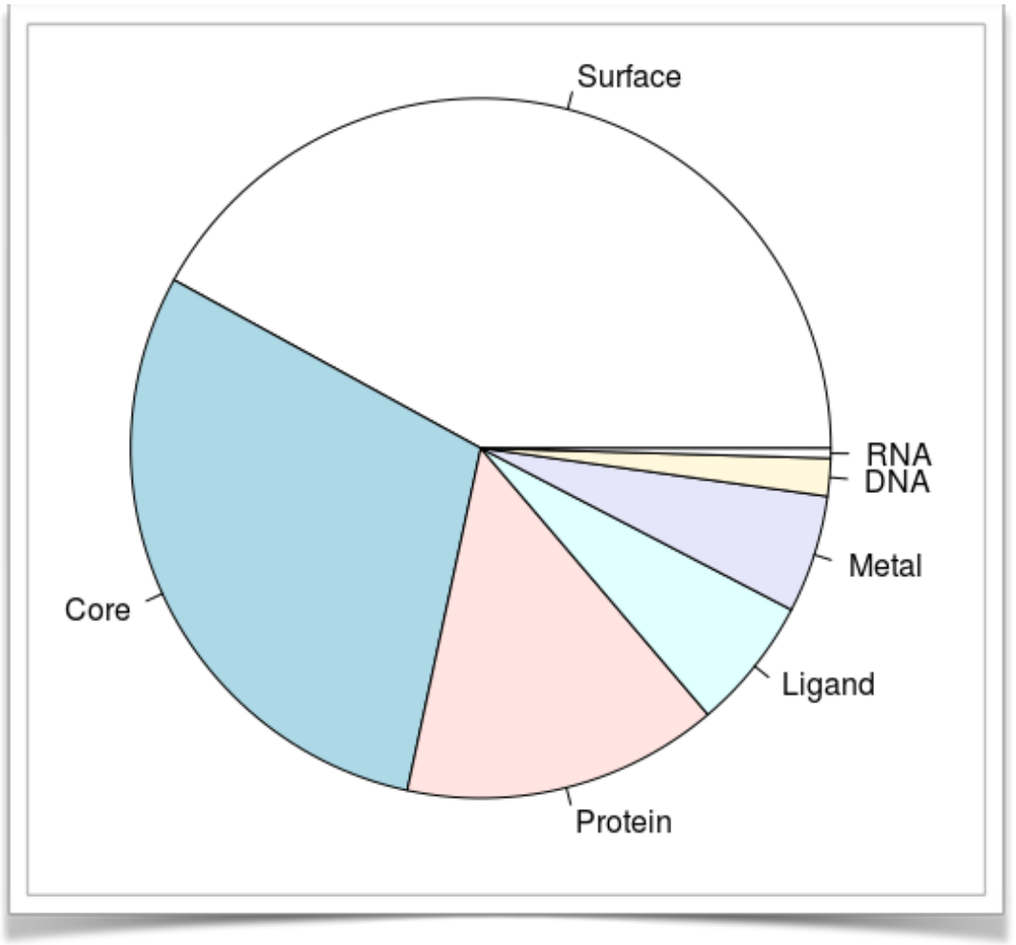
- Every protein position is classified as `Surface`, `Core`, `Ligand`, `Metal`, `Protein`, `DNA`, `RNA`, or `Disorder` (8 categories)
- 23,095,049 rows (1.9 GB)

Pie chart (R)

```
> p <- read.table("proteome.classification.tsv", sep="\t")
> p[ p == "None" ] <- NA
> pp <- p[p$Class <> 'Disorder', ]
> piedata <- aggregate(pp$AC, by=list(Category=pp$Class), FUN=length)
> piedataOrdered <- piedata[ order(-piedata$x), ]
> piedataOrdered
  Category       x
7  Surface 6411178
1     Core 4519347
5  Protein 2228705
3   Ligand  934970
4    Metal  830419
2      DNA  265432
6      RNA   69701

> pie(piedataOrdered$x/nrow(pp),
  labels=piedataOrdered$Category)
```

Pie chart (R)

```
> p <- read.table("proteome.classification.tsv", sep="\t")
> p[ p == "None" ] <- NA
> pp <- p[p$Class <> 'Disorder', ]
> piedata <- aggregate(pp$AC, by=list(Category=pp$Class), FUN=length)
> piedataOrdered <- piedata[ order(-piedata$x), ]
> piedataOrdered
  Category       x
7  Surface 6411178
1     Core 4519347
5  Protein 2228705
3   Ligand  934970
4    Metal  830419
2      DNA  265432
6      RNA   69701

> pie(piedataOrdered$x/nrow(pp),
  labels=piedataOrdered$Category)
```

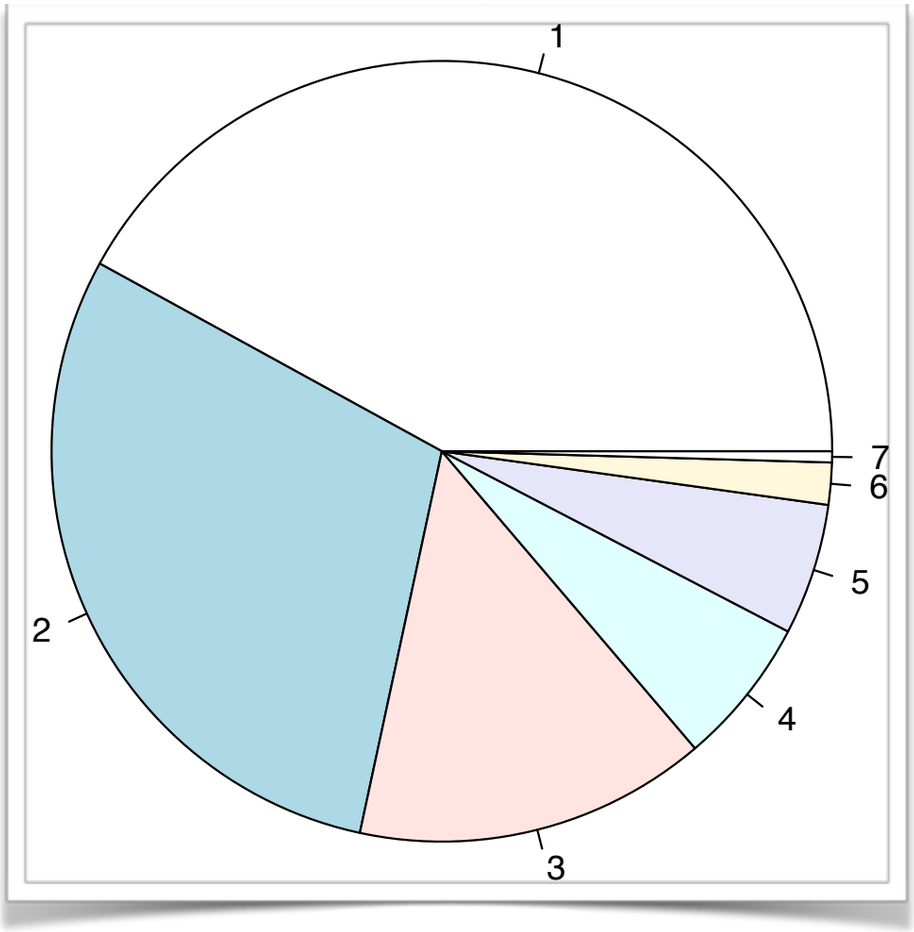


Pie chart (PL/R)

```
kalinina=# CREATE VIEW piechart AS SELECT class, CAST(count(ac) AS float)/(SELECT
count(ac) FROM structman WHERE class <> 'Disorder') AS percentage FROM structman
WHERE class <> 'Disorder' GROUP BY class ORDER BY percentage DESC;
CREATE VIEW

kalinina=# CREATE or REPLACE function r_pie(_float8) RETURNS void AS '
pdf("~/Work/ddG/testpie.pdf")
pie(arg1)
dev.off()
' language 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE pie2pdf (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_pie
);
CREATE AGGREGATE

kalinina=# SELECT pie2pdf(percentage) FROM piechart;
 pie2pdf
---------

(1 row)
```

Pie chart (PL/R)

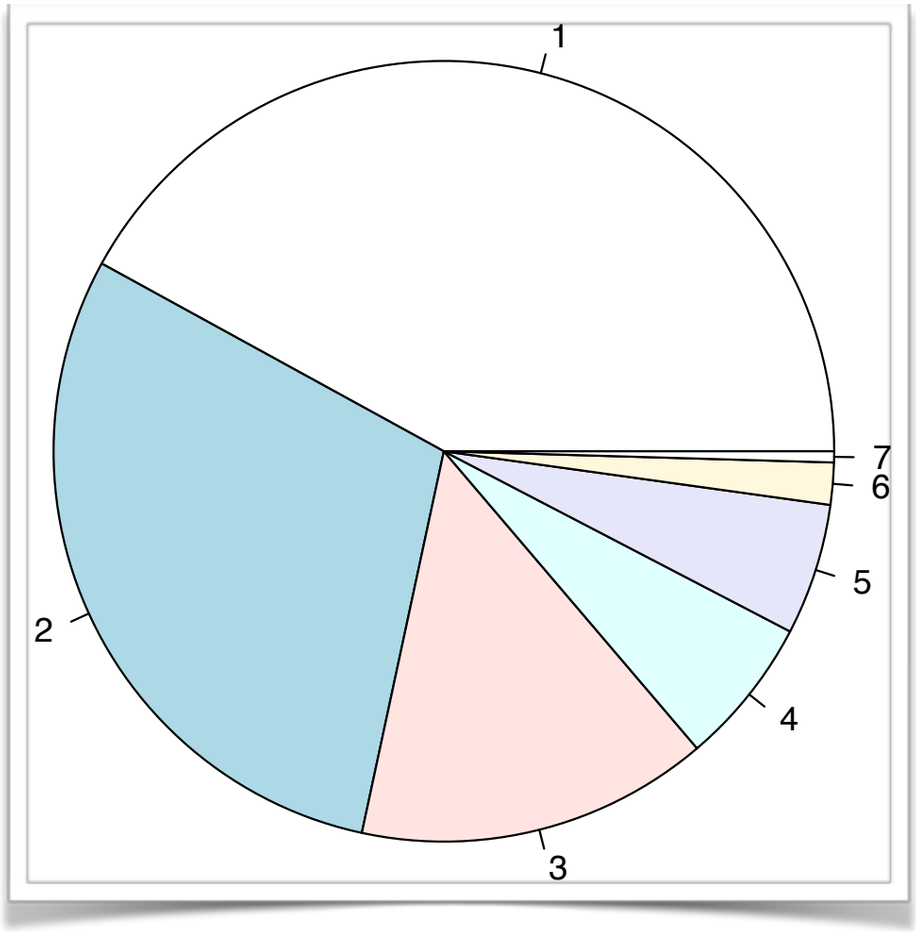
```
kalinina=# CREATE VIEW piechart AS SELECT class, CAST(count(ac) AS float)/(SELECT
count(ac) FROM structman WHERE class <> 'Disorder') AS percentage FROM structman
WHERE class <> 'Disorder' GROUP BY class ORDER BY percentage DESC;
CREATE VIEW

kalinina=# CREATE or REPLACE function r_pie(_float8) RETURNS void AS '
pdf("~/Work/ddG/testpie.pdf")
pie(arg1)
dev.off()
' language 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE pie2pdf (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_pie
);
CREATE AGGREGATE

kalinina=# SELECT pie2pdf(percentage) FROM piechart;
 pie2pdf
---------

(1 row)
```



Pie chart (PL/R)

```
kalinina=# CREATE VIEW piechart AS SELECT class, CAST(count(ac) AS float)/(SELECT
count(ac) FROM structman WHERE class <> 'Disorder') AS percentage FROM structman
WHERE class <> 'Disorder' GROUP BY class ORDER BY percentage DESC;
CREATE VIEW

kalinina=# CREATE or REPLACE function r_pie(_float8) RETURNS void AS '
pdf("~/Work/ddG/testpie.pdf")
pie(arg1)
dev.off()
' language 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE pie2pdf (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_pie
);
CREATE AGGREGATE

kalinina=# SELECT pie2pdf(percentage) FROM piechart;
 pie2pdf
---------

(1 row)
```

No clean solution to pass the names of the categories



Now it starts to pay off

Now it starts to pay off

- `pp` (all rows except ‘Disorder’) has 15,259,752 rows

Now it starts to pay off

- `pp` (all rows except ‘Disorder’) has 15,259,752 rows
- The most expensive command in R:

```
aggregate(pp$AC, by=list(Category=pp$Class), FUN=length)
```

takes ~6.3 sec to execute

Now it starts to pay off

- `pp` (all rows except ‘Disorder’) has 15,259,752 rows
- The most expensive command in R:
  `aggregate(pp$AC, by=list(Category=pp$Class), FUN=length)`
  takes ~6.3 sec to execute
- Selection from `piechart` in the database takes 1.97 sec

Now it starts to pay off

- `pp` (all rows except ‘Disorder’) has 15,259,752 rows
- The most expensive command in R:
  `aggregate(pp$AC, by=list(Category=pp$Class), FUN=length)`
  takes ~6.3 sec to execute
- Selection from `piechart` in the database takes 1.97 sec
- On the other hand, running `median` grouped by `Class` will never finish: full table scan

Statistical significance

- R has implementations of a variety of statistical tests, e.g. Wilcoxon test:

Statistical significance

- R has implementations of a variety of statistical tests, e.g. Wilcoxon test:

```
> wilcox.test(x[x$ClinicalSignificance==‘Pathogenic',]$FX_ddG),
x[x$ClinicalSignificance==‘Benign',]$FX_ddG))

    Wilcoxon rank sum test with continuity correction

data:  x[x$ClinicalSignificance == "Pathogenic", ]$FX_ddG and
x[x$ClinicalSignificance == "Benign", ]$FX_ddG
W = 4419800, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0
```

Statistical significance

- R has implementations of a variety of statistical tests, e.g. Wilcoxon test:

```
> wilcox.test(x[x$ClinicalSignificance==‘Pathogenic',]$FX_ddG),
x[x$ClinicalSignificance==‘Benign',]$FX_ddG))

    Wilcoxon rank sum test with continuity correction

data:  x[x$ClinicalSignificance == "Pathogenic", ]$FX_ddG and
x[x$ClinicalSignificance == "Benign", ]$FX_ddG
W = 4419800, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0

> wilcox.test(x[x$ClinicalSignificance==‘Pathogenic',]$FX_ddG),
x[x$ClinicalSignificance==‘Benign’,]$FX_ddG))$p.value
[1] 1.033810e-167
```

Passing two arrays of datapoint

```
kalinina=# CREATE TABLE ddg (pathogenic float, benign float);
CREATE TABLE
kalinina=# INSERT INTO ddg(pathogenic) SELECT fx_ddg FROM clinvar
WHERE clinicalsignificance = 'Pathogenic';
INSERT 0 20336
kalinina=# INSERT INTO ddg(benign) SELECT fx_ddg FROM clinvar
WHERE clinicalsignificance = 'Benign';
INSERT 0 64090
kalinina=# CREATE TABLE ddg_all (ddg float);
CREATE TABLE
kalinina=# INSERT INTO ddg_all(ddg) SELECT pathogenic FROM ddg;
INSERT 0 84426
kalinina=# INSERT INTO ddg_all(ddg) SELECT benign FROM ddg;
INSERT 0 84426
```

…and calculating statistical significance

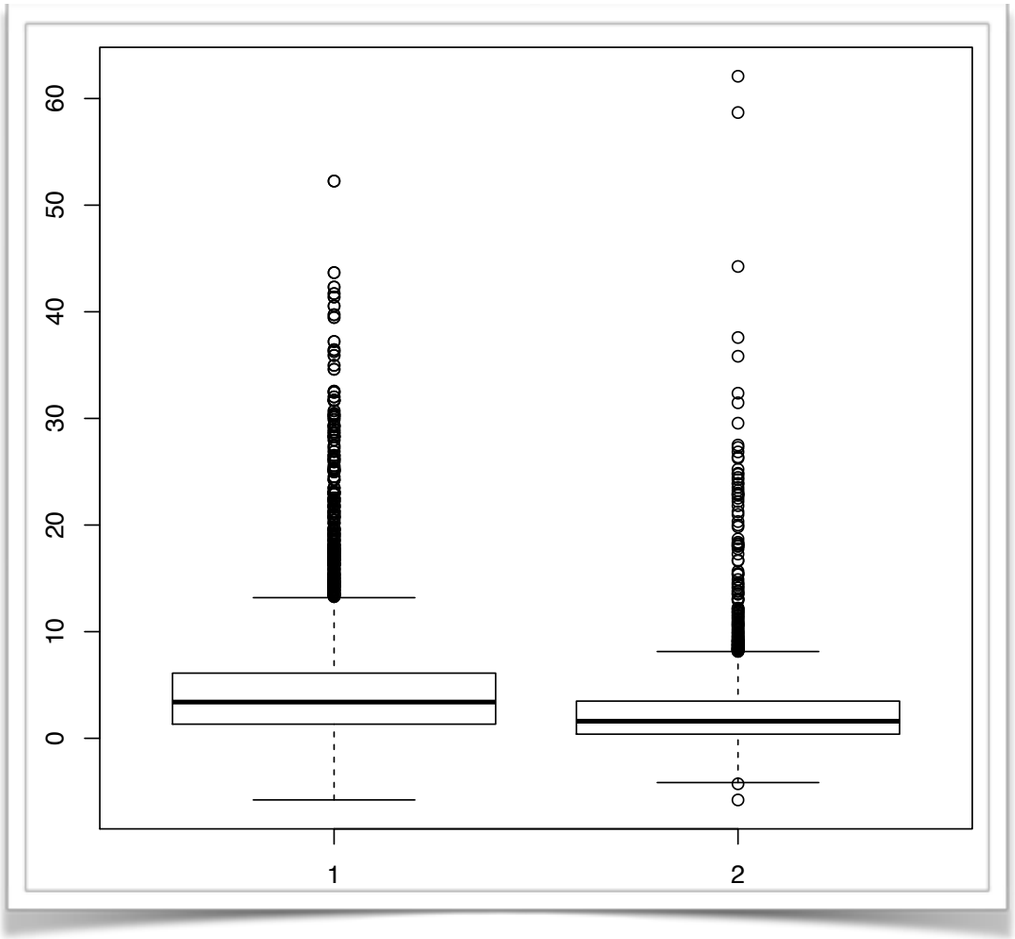
```
kalinina=# CREATE OR REPLACE FUNCTION r_wilcox(_float8) RETURNS float AS
'
x<-arg1[1:length(arg1)/2]
y<-arg1[length(arg1)/2+1:length(arg1)]
wilcox.test(x,y)$p.value
' language 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE wilcox (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_wilcox
);
CREATE AGGREGATE

kalinina=# SELECT wilcox(ddg) FROM ddg_all;
        wilcox
-----------------------
 1.03380966840586e-167
(1 row)
```

…draw plots with two series

```
kalinina=# CREATE OR REPLACE FUNCTION r_plottwo(_float8) RETURNS float AS
'
pdf("testtwo.pdf")
x<-arg1[1:length(arg1)/2]
y<-arg1[length(arg1)/2+1:length(arg1)]
boxplot(x,y)
dev.off()
' language 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE plottwo (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_plottwo
);
CREATE AGGREGATE

kalinina=# SELECT plottwo(ddg) FROM ddg_all;
        plottwo
-----------------------

(1 row)
```

…draw plots with two series

```
kalinina=# CREATE OR REPLACE FUNCTION r_plottwo(_float8) RETURNS float AS
'
pdf("testtwo.pdf")
x<-arg1[1:length(arg1)/2]
y<-arg1[length(arg1)/2+1:length(arg1)]
boxplot(x,y)
dev.off()
' language 'plr';
CREATE FUNCTION

kalinina=# CREATE AGGREGATE plottwo (
sfunc = plr_array_accum,
basetype = float8,
stype = _float8,
finalfunc = r_plottwo
);
CREATE AGGREGATE

kalinina=# SELECT plottwo(ddg) FROM ddg_all;
        plottwo
-----------------------

(1 row)
```



Joins (R)

- Theoretically, you can join in R

Joins (R)

- Theoretically, you can join in R
- Let’s do an inner join:

```
x: chr Gene   ClinicalSignificance uniprot_ac uniprot_pos aa1 aa2 FX_ddG

p: AC Mut  Species Tags  Surface/Core  Class
```

Joins (R)

- Theoretically, you can join in R
- Let's do an inner join:

```
x: chr Gene   ClinicalSignificance uniprot_ac uniprot_pos aa1 aa2 FX_ddG

p: AC Mut  Species Tags  Surface/Core  Class


> library (dplyr)
> joined_data <- t %>% inner_join(p, by = c(c(x$uniprot_ac == p$AC)),
c(x$uniprot_pos == p$Mut)))
Error in Ops.factor(x$uniprot_ac, p$AC) : level sets of factors are
different
```

- You have to have the same set of identifiers in both tables!

Joins (PL/R)

```
kalinina=# SELECT DISTINCT structman.ac AS ac,
clinicalsignificance, fx_ddg INTO core FROM clinvar INNER JOIN
structman ON structman.ac = clinvar.uniprot_ac AND structman.mut
= clinvar.aa1||clinvar.uniprot_pos WHERE structman.class =
'Core';
SELECT 6637

kalinina=# SELECT DISTINCT structman.ac AS ac,
clinicalsignificance, fx_ddg INTO notcore FROM clinvar INNER JOIN
structman ON structman.ac = clinvar.uniprot_ac AND structman.mut
= clinvar.aa1||clinvar.uniprot_pos WHERE structman.class <>
'Core';
SELECT 13430
```

Joins (PL/R)

```
kalinina=# SELECT clinicalsignificance, median(fx_ddg) FROM clinvar GROUP BY
clinicalsignificance;
 clinicalsignificance |  median
----------------------+-----------
 Pathogenic           |    1.7756
 Benign               | 0.6220875
(2 rows)

kalinina=# SELECT clinicalsignificance, median(fx_ddg) FROM core GROUP BY
clinicalsignificance;
 clinicalsignificance | median
----------------------+---------
 Pathogenic           |  3.4113
 Benign               | 1.55485
(2 rows)

kalinina=# SELECT clinicalsignificance, median(fx_ddg) FROM notcore GROUP BY
clinicalsignificance;
 clinicalsignificance |  median
----------------------+----------
 Pathogenic           | 1.003565
 Benign               | 0.424089
(2 rows)
```

Summary

Summary

- Data analysis can be done with PL/R (almost) as easily as in the R environment

Summary

- Data analysis can be done with PL/R (almost) as easily as in the R environment
- Syntax is more cumbersome

Summary

- Data analysis can be done with PL/R (almost) as easily as in the R environment
- Syntax is more cumbersome
- Passing two arrays of datapoints is a problem

Summary

- Data analysis can be done with PL/R (almost) as easily as in the R environment
- Syntax is more cumbersome
- Passing two arrays of datapoints is a problem
- However, one can benefit from data handling in the database

Summary

- Data analysis can be done with PL/R (almost) as easily as in the R environment
- Syntax is more cumbersome
- Passing two arrays of datapoints is a problem
- However, one can benefit from data handling in the database
- **Feedback**: https://2019.fosdempgday.org/f