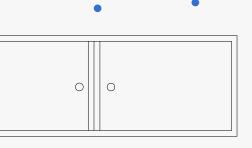
Diagnóstico prenatal de enfermedades genéticas usando Python

Marina Moro López











whoami

- Ingeniera biomédica
- Futura doctora en biomedicina
- 'Programadora' en el ámbito científico

Índice

¿Qué es el diagnóstico prenatal?

Caso práctico
de diagnóstico
prenatal

Conceptos básicos de genética

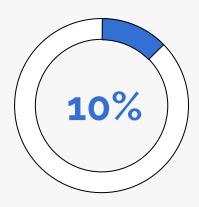
Más aplicacionesclínicas congenética y Python



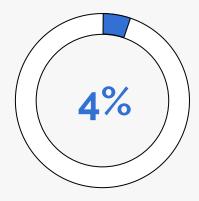
1 ¿Qué es el diagnóstico prenatal?

Contexto

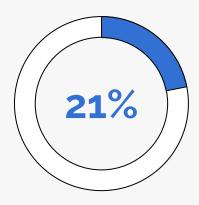
Existen alrededor de 6.000 enfermedades genéticas



Población que sufre una enfermedad genética a lo largo de su vida



Neonatos con una enfermedad genética



Muertes de neonatos debido a enfermedades genéticas

Identificación de enfermedades genéticas antes del nacimiento

Por edad o mutaciones de los progenitores, enfermedades genéticas en hijxs existentes o indicios en los resultados de otras pruebas clínicas

Identificación de enfermedades genéticas antes del nacimiento

Por edad o mutaciones de los progenitores, enfermedades genéticas en hijxs existentes o indicios en los resultados de otras pruebas clínicas

	ltrasc	MIMA
u	'LLI ast	nnao

Análisis hormonal

ADN fetal

NO INVASIVO

Defectos a simple vista

Hormonas en la sangre de la persona gestante

Mutaciones del ADN

Identificación de enfermedades genéticas antes del nacimiento

Por edad o mutaciones de los progenitores, enfermedades genéticas en hijxs existentes o indicios en los resultados de otras pruebas clínicas

	Ultrasonido	Análisis hormonal	ADN fetal
NO INVASIVO	Defectos a simple vista	Hormonas en la sangre de la persona gestante	Mutaciones del ADN
INVASIVO	Amniocentesis	Vellosidades coriónicas	Biopsia de placenta
	Defectos en cromosomas	Defectos en cromosomas y mutaciones del ADN	Defectos en cromosomas

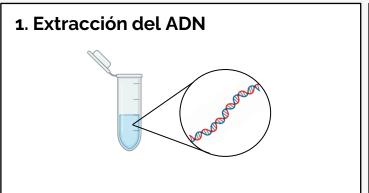
Identificación de enfermedades genéticas antes del nacimiento

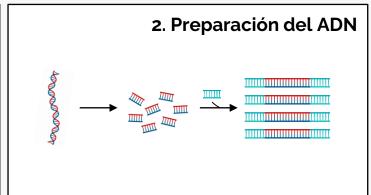
Por edad o mutaciones de los progenitores, enfermedades genéticas en hijxs existentes o indicios en los resultados de otras pruebas clínicas

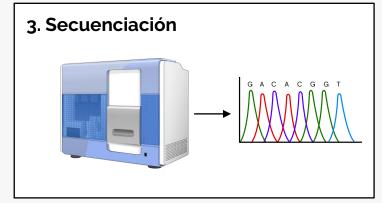
	Ultrasonido	Análisis hormonal	ADN fetal
NO INVASIVO	Defectos a simple vista	Hormonas en la sangre de la persona gestante	Mutaciones del ADN
INVASIVO	Amniocentesis	Vellosidades coriónicas	Biopsia de placenta
	Defectos en cromosomas	Defectos en cromosomas y mutaciones del ADN	Defectos en cromosomas

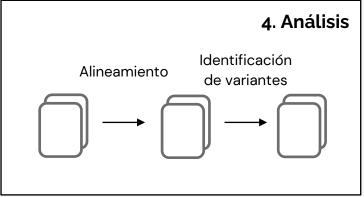
^{*} Diagnóstico preimplantacional: análisis del material genético del embrión antes de implantarlo in vitro

Workflow











2 Conceptos básicos de genética

Cromosoma - gen - ADN

Estructura que contiene todos los genes



Cromosoma - gen - ADN

Estructura que contiene todos los genes

Segmento de ADN que determina un rasgo





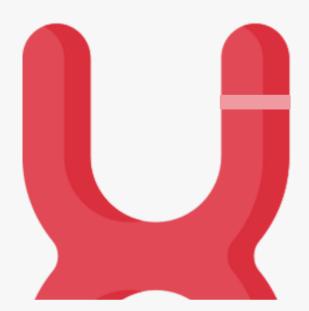
Cromosoma - gen - ADN

Estructura que contiene todos los genes

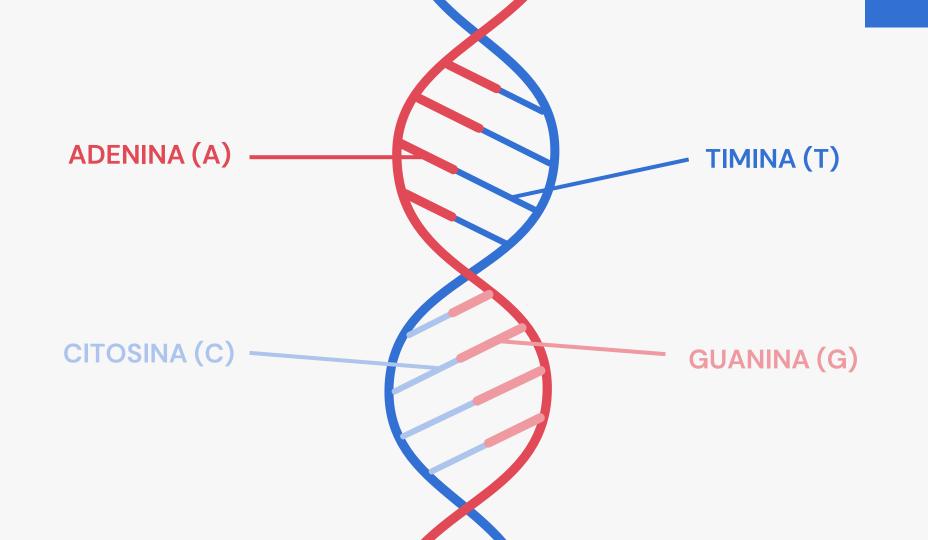
Segmento de ADN que determina un rasgo

Doble hélice formada por bases

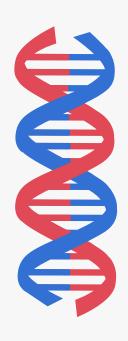




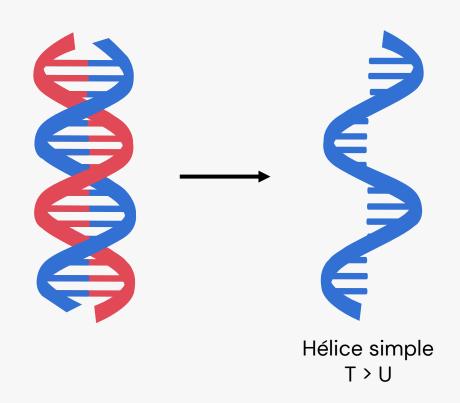




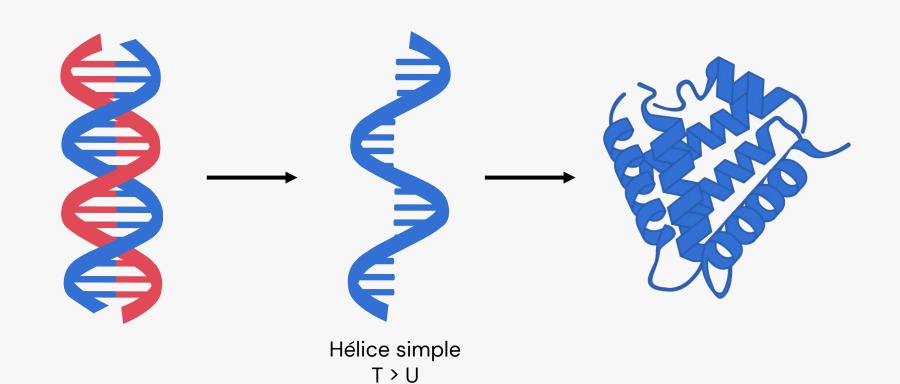
ADN - ARN - proteína



ADN - ARN - proteína



ADN - ARN - proteína





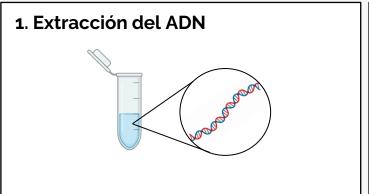
3 Caso práctico de diagnóstico prenatal

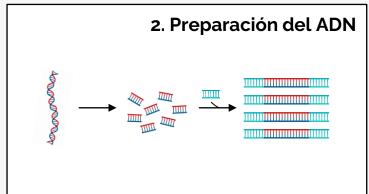
Epidermólisis bullosa

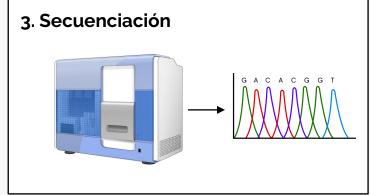
- Enfermedad genética rara (1 de cada 60.000 recién nacidos) que afecta a las mucosas y la piel
- Provoca fragilidad extrema en mucosas y piel, apareciendo ampollas con el contacto o incluso de forma espontánea (piel de mariposa)
- Otras complicaciones son lesiones dentales, atrofia muscular, estrechamiento de las vías respiratorias, gastrointestinales y urogenitales, y cáncer

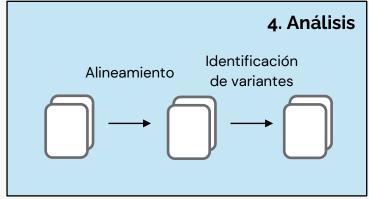
Causada por mutaciones en genes de proteínas de la piel, como colágeno VII, lamininas, integrinas...

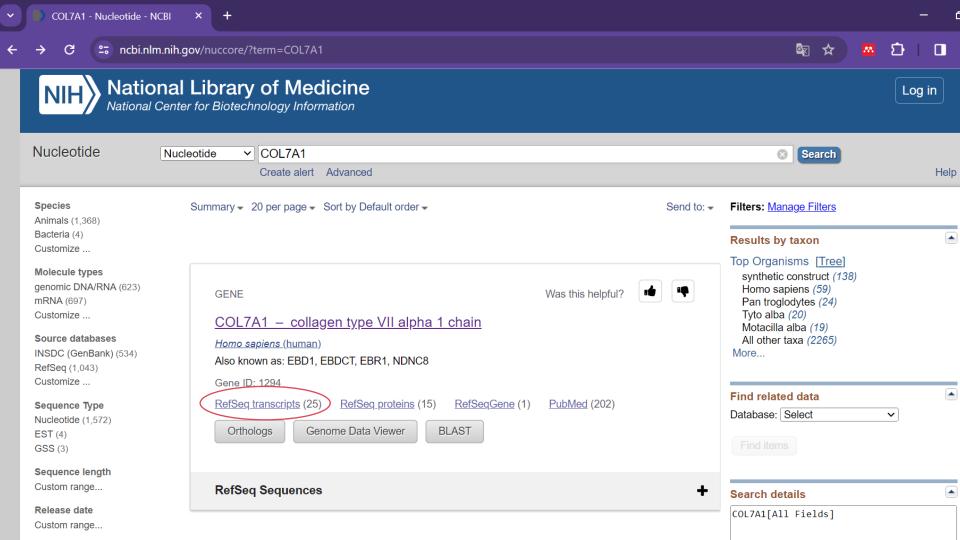
Workflow

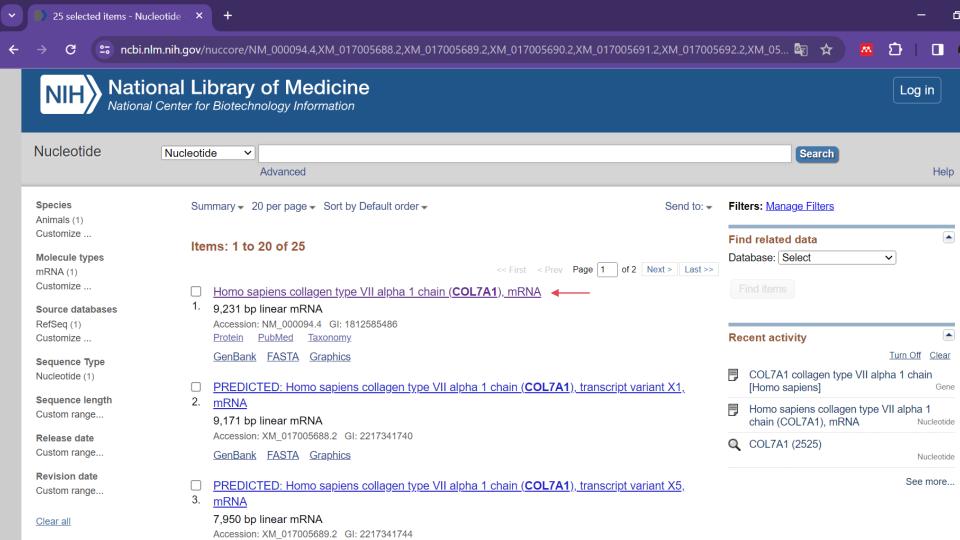


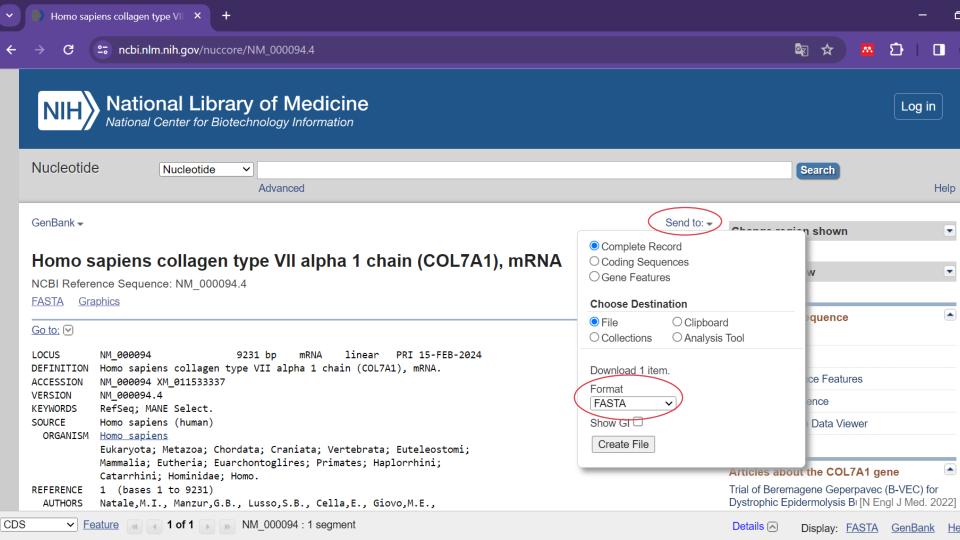












```
from tkinter.filedialog import askopenfile
def main():
   gene file = askopenfile(mode='r')
    gene_seq = gene_file.readlines()[1:]
   gene_seq = ''.join(gene_seq).replace('\n', '')
    patient file = askopenfile(mode='r')
    patient seq = patient file.readlines()[1:]
    patient seq = ''.join(patient seq).replace('\n', '')
    seq
                 marinamorolopez / alignment-WTMPython
        seq_arranmenc.appenu(pos_arran)
   mutation_pos = [i for i, val in enumerate(seq_alignment) if not val]
    for i in range(len(mutation_pos)):
       original bases = gene seq[mutation_pos[i]]
       mutated bases = patient seg[mutation pos[i]]
   mutation_pos_corrected = [x+1 for x in mutation_pos]
    print ("Mutation positions: " + str(mutation pos corrected))
    print("Original bases: " + original_bases)
    print("Mutated bases: " + mutated bases)
```

main()

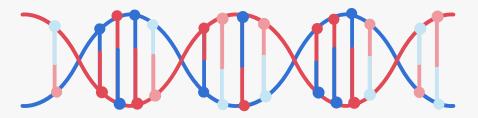
```
def main():
    gene file = askopenfile(mode='r')
    gene_seq = gene_file.readlines()[1:]
    gene_seq = ''.join(gene_seq).replace('\n', '')
    patient file = askopenfile(mode='r')
    patient seq = patient file.readlines()[1:]
    patient seq = ''.join(patient seq).replace('\n', '')
    seq alignment = []
    for i in range(len(gene seq)):
        pos_align = (gene_seq[i]==patient_seq[i])
        seq alignment.append(pos align)
   mutation_pos = [i for i, val in enumerate(seq_alignment) if not val]
    for i in range(len(mutation_pos)):
        original bases = gene seq[mutation_pos[i]]
        mutated bases = patient seg[mutation pos[i]]
    mutation_pos_corrected = [x+1 for x in mutation_pos]
    print ("Mutation positions: " + str(mutation pos corrected))
    print("Original bases: " + original_bases)
    print("Mutated bases: " + mutated bases)
main()
```

from tkinter.filedialog import askopenfile

GEN COL7A1 ORIGINAL



GEN COL7A1 MUTADO



GEN COL7A1 ORIGINAL GEN COL7A1 MUTADO

TCTCTGGAGAACAGGGACCCCCTGGACTCAAGGGTGCTAAGGGGGAGCCGGGCAGCAATGGTGACCAAGG
TCCCAAAGGAGACAGGGGTGTGCCAGGCATCAAAGGAGACCGGGGAGAGCCTGGACCGAGGGTCAGGAC
GGCAACCCGGGTCTACCAGGAGAGCGTGGTATGGCTGGGCCTGAAGGGAAGCCGGGTCTGCAGGGTCCAA
GAGGCCCCCCTGGCCCAGTGGGTGGTCATGGAGACCCTGGACCACCTGGTGCCCCGGGTCTTGCTGGCCC
TGCAGGACCCCAAGGACCTTCTGGCCTGAAGGGGAGCCTGGAGAGACAGGACCTCCAGGACGGGCCTG
ACTGGACCTACTGGAGCTGTGGGACTTCCTGGACCCCCCGGCCCTTCAGGCCTTGTGGGTCCACAGGGGT
CTCCAGGTTTGCCTGGACAAGTGGGGGAGACAGGGAAGCCCCAGGTCGAGATGGTGCCAGTGG

TCTCTGGAGAACAGGGACCCCCTGGACTCAAGGGTGCTAAGGGGGAGCCGGGCAGCAATGGTGACCAAGG
TCCCAAAGGAGACAGGGGTGTGCCAGGCATCAAAGGAGACCGGGGAGAGCCTGGACCGAGGGTCAGGAC
GGCAACCCGGGTCTACCAGGAGAGCGTGGTATGGCTGGGCCTGAAGGGAAGCCGGGTCTGCAGGGTCCAA
GAGGCCCCCCGGGCCCAGTGGGTGGTCATGGAGACCCTGGACCACCTGGTGCCCCGGGTCTTGCTGGCCC
TGCAGGACCCCAAGGACCTTCTGGCCTGAAGGGGAGCCTGGAGAGACAGGACCTCCAGGACGGGCCTG
ACTGGACCTACTGGAGCTGTGGGACTTCCTGGACCCCCCGGCCCTTCAGGCCTTGTGGGTCCACAGGGGT
CTCCAGGTTTGCCTGGACAAGTGGGGAAACAGGAACCCCCAGGTCGAGATGGTGCCAGTGG

TCTCTGGAGAACAGGGACCCCCTGGACTCAAGGGTGCTAAGGGGGAGCCGGGCAGCAATGGTGACCAAGG TCCCAAAGGAGACAGGGTGTGCCAGGCATCAAAGGAGACCGGGGGAGAGCCTGGACCGAGGGGTCAGGAC GGCAACCCGGGTCTACCAGGAGAGCGTGGTATGGCTGGGCCTGAAGGGAAGCCGGGTCTGCAGGGTCCAA GAGGCCCCCCTGGCCCAGTGGGTGGTCATGGAGACCCTGGACCACCTGGTGCCCCGGGTCTTGCTGGCCC TGCAGGACCCCAAGGACCTTCTGGCCTGAAGGGGGGGCCTGGAGAGACAGGACCTCCAGGACGGGCCTG ACTGGACCTACTGGAGCTGTGG GGCCTTGTGGGTCCACAGGGGT CTCCAGGTTTGCCTGGACAAGT Mutation positions: [6591]
Original bases: T CAGGTCGAGATGGTGCCAGTGG TCTCTGGAGAACAGGGACCCCC Mutated bases: C CGGGCAGCAATGGTGACCAAGG TCCCAAAGGAGACAGGGGTGTGCCAAAGCATCAAAGGAGACCGAGGGGTCAGGAC GGCAACCCGGGTCTACCAGGAGAGCGTGGTATGGCTGGGCCTGAAGGGAAGCCGGGTCTGCAGGGTCCAA GAGGCCCCCCCGGGCCCAGTGGGTGGTCATGGAGACCCTGGACCACCTGGTGCCCCGGGTCTTGCTGGCCC TGCAGGACCCCAAGGACCTTCTGGCCTGAAGGGGGAGCCTGGAGAGACAGGACCTCCAGGACGGGGCCTG ACTGGACCTACTGGAGCTGTGGGACTTCCTGGACCCCCGGCCCTTCAGGCCTTGTGGGTCCACAGGGGT CTCCAGGTTTGCCTGGACAAGTGGGGAGACAGGGAAGCCGGGAGCCCCAGGTCGAGATGGTGCCAGTGG



4

Más aplicaciones clínicas con genética y Python

Más aplicaciones

Descripción de nuevas mutaciones

Diseño de tratamientos genéticos

Estudio de eficacia de medicamentos

Predicción de efectos de mutaciones

Análisis de big data clínico

iGracias!:)

¿Preguntas?



marinaml5598@gmail.com



marinamorolopez



ീപ്ര Marina Moro López



marinamorolopez / alignment-WTMPython



