24/04/2019

Endemic peruVIAN species application project

LUIS RAMOS GRADOS, VICTOR PONCE PINEDO, CARLOS ESPINOZA MANSILLA | biología computacional | PROFESOR: ciro nuñez

1. Introducción:

Es necesario mantener un registro de todas las especies que pueblan la tierra, que científicos puedan analizar en el presente y futuro con vista a comprender la diversa fauna de nuestro planeta, en este proyecto queremos aportar nuestro pequeño grano de arena a esta causa haciendo un registro del árbol filogenético de las especies endémicas en nuestro país.

1. Propuesta:

Nuestro proyecto consistirá de una base de datos en la que estará la información genética de especies endémicas de Perú accesibles a través de una aplicación hecha en Python tkinter, dichos datos serán llamados desde la librería Bio Python y se mostrarán al usuario. Mediante la estructura:

Cabe recalcar que vamos a trabajar con 10 aves peruanas y analizaremos la proteína llamada citocromo b. Su función es explicada en el inciso 4

Como se mencionó antes, el programa será codificado en Python y nos permitirá realizar alineamientos múltiples tanto de genes como de proteínas y esto se realizará mediante Biopython y MUSCLE. Además, para la generación del árbol filogenético usaremos Biopython, el cual nos brinda la posibilidad de generar nuestros árboles a partir del método UPGMA y usando una matriz de distancias.

1. Cronograma:

| Tarea | Fecha de vencimiento |
| --- | --- |
| Entrega de la propuesta de proyecto | 29/04/2019 |
| Definir: Especies, Genes, y Proteínas dentro de una base de datos. | 13/05/2019 |
| Versión final de la propuesta en PDF y demostración al 30% | 20/05/2019 |
| Realizar alineamientos de genes y proteínas en la base de datos con el código Biopython | 27/05/19 |
| Diseño de la interfaz gráfica | 04/06/2019 |
| Implementación de la GUI y demostración al 70% | 12/06/2019 |
| Generación de Árboles Filogenéticos a partir de la información generada. | 17/06/2019 |
| Demostracion al 100% incluyendo Codigo fuente, Librerias, datos de prueba, reporte final. En formato digital (DVD) | 25/06/2019 |

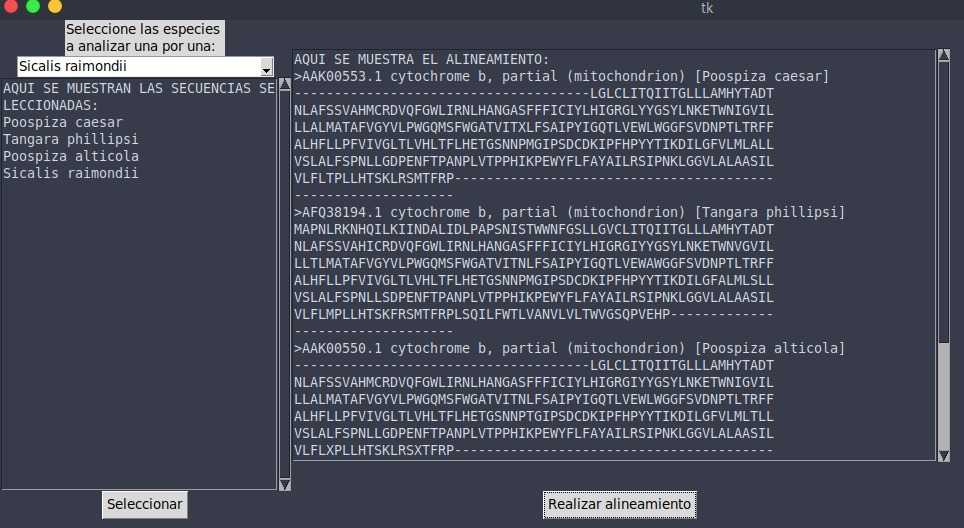
1. Desarrollo final del proyecto:
   1. Objetivos

* Desarrollar un programa capaz de identificar, alinear y crear árboles filogenéticos a partir de secuencias genéticas de aves oriundas de Perú.
  1. Descripción del proyecto
     1. Proteina estudiada
        + Citocromo b: ¿Por qué?
          - el citocromo b es un componente del [complejo III](https://es.wikipedia.org/wiki/Coenzima_Q_citocromo_c_reductasa) de la [cadena respiratoria](https://es.wikipedia.org/wiki/Cadena_respiratoria) , también conocido como complejo bc1 o ubiquinol-citocromo c reductasa. En los [cloroplastos](https://es.wikipedia.org/wiki/Cloroplasto) y [cianobacterias](https://es.wikipedia.org/wiki/Cianobacteria), existe una proteína análoga llamada citocromo b6, un componente de la plastoquinona-plastocianín reductasa, también conocida como complejo b6f. Estos complejos se encuentran involucrados en el transporte de electrones y la generación de [ATP](https://es.wikipedia.org/wiki/Adenosín_trifosfato) y por lo tanto juegan un rol vital en las células.
          - Escogimos tomar secuencias pertenecientes a esta en particular porque la región del [ADN mitocondrial](https://es.wikipedia.org/wiki/ADN_mitocondrial) que codifica para el citocromo b es comúnmente utilizada para determinar las relaciones [filogenéticas](https://es.wikipedia.org/wiki/Filogenética) entre organismos sobre la base de la variabilidad de su secuencia. Es considerada la más útil para determinar relaciones entre [familias](https://es.wikipedia.org/wiki/Familia_(biología)) y [géneros](https://es.wikipedia.org/wiki/Genus). Los estudios comparativos que involucran al citocromo b han resultado en nuevos esquemas de clasificación y han sido utilizados para asignar las nuevas descripciones de especies a los géneros conocidos, como así también para profundizar la comprensión de las relaciones evolutivas.
     2. Especies analizadas:
        + Iridosornis reinhardti
        + Sicalis raimondii
        + Poospiza caesar
        + Poospiza rubecula
        + Poospiza alticola
        + Incaspiza ortizi
        + Incaspiza personata
        + Incaspiza pulchra
        + Piezorhina cinerea
        + Tangara phillipsi
     3. Herramientas utilizadas
        + - Muscle

MUSCLE es uno de los programas de alineación múltiple de mejor rendimiento según las pruebas de referencia publicadas, con una precisión y una velocidad que son siempre mejores que CLUSTALW. MUSCLE puede alinear cientos de secuencias en segundos. La mayoría de los usuarios aprenden todo lo que necesitan saber sobre MUSCLE en unos pocos minutos: solo se necesitan unas pocas opciones de línea de comandos para realizar tareas comunes de alineación.

* + - * + api de Wikipedia

permite obtener información escrita de las especies indicadas directamente desde la enciclopedia virtual.

* + - * Lenguaje
        + Este proyecto ha sido realizado en Python.
      * Librerias usadas
        + Biopython: son una serie de aplicaciones y programas informáticos pensados para cuantificar y hacer cálculos con datos biológicos, programados por una comunidad internacional. Su objetivo principal es el de desarrollar el mayor número posible de [bibliotecas informáticas](https://es.wikipedia.org/wiki/Biblioteca_informática) basadas en el lenguaje de programación [Python](https://es.wikipedia.org/wiki/Python), que tengan aplicaciones [bioinformáticas](https://es.wikipedia.org/wiki/Bioinformática) y que estén disponibles para un público lo más amplio posible.
        + Tkinter: Tkinter es un [binding](https://es.wikipedia.org/wiki/Binding) de la [biblioteca](https://es.wikipedia.org/wiki/Biblioteca_(informática)) gráfica [Tcl/Tk](https://es.wikipedia.org/wiki/Tcl/Tk) para el [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programación) [Python](https://es.wikipedia.org/wiki/Python). Se considera un estándar para la [interfaz gráfica de usuario](https://es.wikipedia.org/wiki/Interfaz_gráfica_de_usuario) (GUI) para Python.
  1. Algoritmos e implementación computacional
     1. Alineamiento múltiple de secuencia
        + El algoritmo MUSCLE se desarrolla en tres etapas: el borrador progresivo, progresivo mejorado y refinamiento. En el proyecto de etapa progresiva, el algoritmo produce un borrador de alineación múltiple, enfatizando la velocidad sobre la precisión. En la etapa progresiva mejorada, la distancia de Kimura se usa para volver a estimar el árbol binario utilizado para crear la alineación de tiro, lo que a su vez produce una alineación múltiple más precisa. La etapa final de refinamiento refina la alineación mejorada realizada en el paso dos. Las alineaciones múltiples están disponibles al final de cada etapa. En las primeras dos etapas del algoritmo, la complejidad del tiempo es O(N2L + NL2), la complejidad del espacio es O(N2 + NL + L2). La etapa de refinamiento agrega a la complejidad del tiempo otro término, O(N3L). MUSCLE se usa a menudo como un reemplazo para Clustal, ya que generalmente (pero no siempre) da mejores alineamientos de secuencia, dependiendo de las opciones elegidas. Además, MUSCLE es significativamente más rápido que Clustal, más aún para alineaciones más grandes.
     2. Generación de árboles filogenéticos:
        + Matriz de distancias
          - La matriz de distancias se calcula a partir de un alineamiento multiple de secuencias, en el cual una fila o columna representa una secuencia y cada celda es igual a la cantidad de diferencias entre un par de secuencias.
        + Árbol UPGMA (Unweighted Pair-Group Method with Arithmetic)
          - Es un método que procede por agrupación de las secuencias que presentan la menor distancia genética. La agrupación de las dos secuencias más relacionadas produce el primer [nodo](https://es.wikipedia.org/wiki/Nodo), que se incorpora a una nueva matriz en la que se calcula su distancia a cada una de las secuencias restantes como la media aritmética de las distancias de las dos secuencias componentes del nodo. El proceso se repite hasta que todas las secuencias quedan agrupadas y unidas por nodos internos.
        + Arbol NJ (Neighbor-joining)
          - Es un método basado en el criterio de mínima evolución (BME: balanced minimum evolution), en el que el mejor árbol es aquel que minimiza la longitud de las ramas internas. Para ello, a partir de un árbol en estrella, se determina la pareja de secuencias más cercanas y se unen mediante un nodo interno. Este proceso se repite con el resto de secuencias hasta que quedan todas unidas por nodos internos que minimizan la longitud de cada una de las ramas internas, aunque no se asegura una longitud mínima global.
  2. Resultados
     1. Screenshots de la GUI:
  3. Conclusiones
     1. Efectividad del trabajo realizado:
        + nuestra GUI facilita enormemente el trabajo de alinear secuencias con un solo botón sin necesidad de saber programar, igualmente es capaz de crear árboles filogenéticos, además de ser altamente eficiente en tiempo y rendimiento gracias a la herramienta MUSCLE y a los algoritmos de generación de árboles.
  4. Referencias
     1. manuales o páginas web usados para el trabajo:

<https://es.wikipedia.org/wiki/Filogen%C3%A9tica_computacional#M%C3%A9todos_de_matriz_de_distancias>

<https://www.drive5.com/muscle/>

* + 1. Documentación/info de biopython:

<https://biopython.org/wiki/Documentation>

* + 1. Documentación/info de tkinter:

<http://tkinter.fdex.eu/>