



Nombre: Paulo Gonzalez

Objetivo:

- Consolidar los conocimientos adquiridos en clase para desarrollar simulaciones de eventos discretos.

Introducción:

El golpe económico de la crisis sanitaria del corona virus no va a ser cosa de semanas, sino de meses. Dentro de una de las etapas importantes posteriores a las elecciones presidenciales son el análisis y tendencia que tiene el actual presidente. Para ello se plantea realizar un sistema de regresión que permita identificar cual es la tendencia en base al manejo de las redes sociales (Twitter y/o Facebook) del presidente [1].

Las regresiones lineales pueden aprender por sí mismos y en este caso obtener automáticamente esa “recta o ecuación” que buscamos con la tendencia de predicción. Para ello se puede medir el error con respecto a los puntos de entrada y el valor “Y” de salida real [3].

Enunciado:

- Diseñe y desarrolle un modelo y/o script que permita simular el siguiente caso real:
 - Obtener datos de tendencia de twitter o facebook(crawler o webscraping), para ello se puede obtener a través del API [4].
 - **Title:** Titulo del Post/Twitter
 - **Word count:** la cantidad de palabras del artículo,
 - **# of Links:** los enlaces externos que contiene,
 - **# of comments:** cantidad de comentarios,
 - **# Shares:** compartidos.
 - **HashTag**
 - Etc.

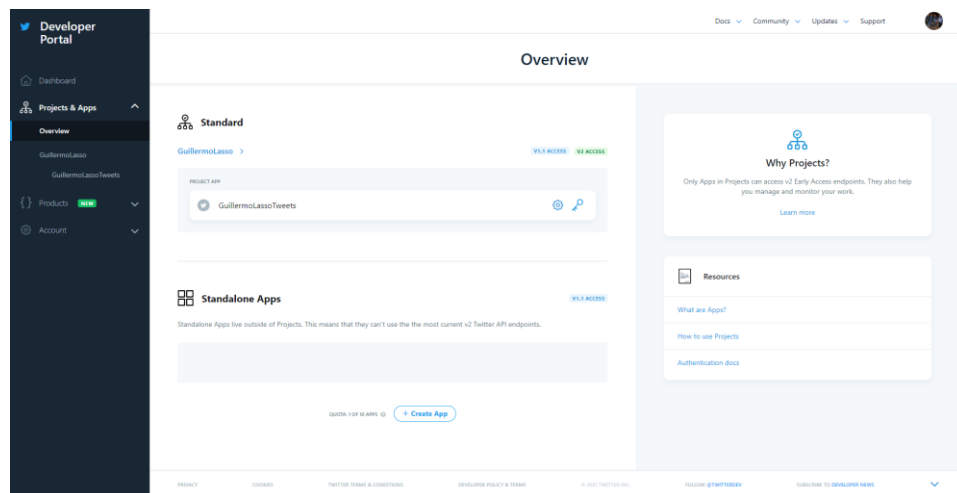
En base a ello, se pretende proponer y generar una predicción de cuántas veces será compartido un post/twitter utilizando regresión [2].



DESARROLLO OBTENCION DE TWEETS (PUNTO 1)

1. Obtención de Tweets

Para este punto se trabajo con la API que nos proporciona twitter; para ello generamos una cuenta de desarrollador en donde nos registraremos y generaremos una aplicación, la cual nos estrega las keys y el token para realizar el consumo de la API y obtener todos los tweets relacionados con Guillermo Lasso



Una vez que obtenemos esa información trabajaremos con la librería Tweepy que nos permite acceder y consumir la API de Twitter, con la cual generaremos una función para obtener los tweets de Guillermo Lasso y con ello generar un CSV que nos servirá para construir un modelo de regresión lineal y polinomial; cabe recalcar que Tweepy nos entrega información como un JSON, por lo cual lo almacenaremos en un archivo .TXT que luego convertiremos a CSV.

```
#importamos la libreria y json para poder trabajar con la informaicon que obtenemos
import tweepy,json

access_token="471944707-4Yt2hcsrdnDbbs3K5uUDg1LW6ZPSMt9BLUdyRlix"
access_token_secret="JT1QMhK20INz9yorTeXi0039whdAuxZRWNq00H5EhVVpt"
consumer_key="KQJ3Lc1BPf0gWkWBGBMiHujkl"
consumer_secret="JwZwyeeE1xe5i0EUIulGiR34i1ITRuejs1XjYxCMPjGxYq17j0"
auth= tweepy.OAuthHandler(consumer_key,consumer_secret)
auth.set_access_token(access_token,access_token_secret)
max_tweets=1000
```

Debemos establecer nuestros tokens y key para acceder a la API, además estableceremos el máximo de tweets a ser consumidos.

Ahora generamos un método en el cual obtendremos la información de los tweets además generaremos el CSV, esto ya que obtenemos un .txt que lo podemos transformar en un .csv, tweets[] será el arreglo que contemplará y almacenar cada tweet que este relacionado con Guillermo Lasso



Simulación

Tema: Simulación de Eventos Examen.

Este método llamará la API de twitter, la cual se encargará de devolvernos información según el filtro que le hemos puesto, en este caso el método hará un llamado a la API y por medio de un for recorrerá y almacenar en un JSON toda la información relacionada con Guillermo Lasso.

```
tweets=[]
class GetTweets(tweepy.StreamListener):
    def __init__(self,api=None):
        super(GetTweets,self).__init__()
        self.tweets=0
        self.file=open("tweets.txt","w")
    def on_status(self,status):
        tweet=status._json
        self.file.write(json.dumps(tweet)+'\n')
        tweet_list.append(status)
        self.tweets+=1
        if self.tweets<max_tweets:
            return True
        else:
            return False
        self.file.close()
l = GetTweets()
stream =tweepy.Stream(auth,l)
#En esta linea filtramos la informacion que vamos a obtener
stream.filter(track=['LassoGuillermo'])
```

2. Lectura y limpieza de datos

Una vez obtenido el TXT procedemos a transformar a un CSV y sus datos en string ya que con ello podemos trabajar de una mejor manera.

```
In [4]: #Leemos los datos del csv generado
df_tweetsLasso=pd.read_csv('Lasso_tweets.csv', index_col=False)
df_tweetsLasso = df_tweetsLasso[df_tweetsLasso.Text.notna()]
df_tweetsLasso.head(20)
```

Out[4]:

	UserName	Handle	Timestamp	Text	Comments	Likes	Retweets
0	Guillermo Lasso	@LassoGuillermo	2021-05-29T21:35:01.000Z	¡Felicidades \n@MorejonGlenda!\n Ella y más de...	24	153	762
1	Guillermo Lasso	@LassoGuillermo	2021-05-28T23:57:43.000Z	¡Una buena noticia!\n@EmbajadaChinaEc\n me inf...	628	1,6 mil	10,5 mil
2	Guillermo Lasso	@LassoGuillermo	2021-05-28T22:23:22.000Z	¡Primera visita a Guayas! Nos reunimos en la G...	134	398	2 mil
3	Guillermo Lasso	@LassoGuillermo	2021-05-26T16:59:31.000Z	Los invito a acompañarnos en este emotivo even...	108	293	1,1 mil
4	Guillermo Lasso	@LassoGuillermo	2021-05-26T15:37:41.000Z	FAMILIA \n\n#JuntosLoLogramos	703	1,4 mil	16,7 mil
5	Guillermo Lasso	@LassoGuillermo	2021-05-26T13:40:49.000Z	Nuestro proyecto de Ley Orgánica de Libre Expr...	268	599	2 mil
6	Guillermo Lasso	@LassoGuillermo	2021-05-26T04:32:57.000Z	Hoy envié a la Asamblea Nacional el proyecto d...	498	1 mil	5 mil
7	Presidencia ECU	@Presidencia_Ec	2021-05-25T21:48:41.000Z	El presidente \n@LassoGuillermo\n, durante su ...	167	558	4,5 mil
8	Guillermo Lasso	@LassoGuillermo	2021-05-25T20:04:59.000Z	Hoy tuve la oportunidad de dialogar con \n@leo...	580	878	4,3 mil
9	Guillermo Lasso	@LassoGuillermo	2021-05-25T16:16:52.000Z	Fue un honor recibir a representantes de \n@US...	207	638	2,9 mil
10	Guillermo Lasso	@LassoGuillermo	2021-05-25T03:11:40.000Z	Esta noche mantuvimos la primera reunión del C...	368	1,1 mil	5,8 mil
11	Guillermo Lasso	@LassoGuillermo	2021-05-24T19:11:36.000Z	Tengo el corazón lleno de gratitud, cariño y c...	210	966	5,3 mil
12	Comunicación Ecuador	@ComunicacionEc	2021-05-24T18:39:24.000Z	AHORA El presidente \n@LassoGuillermo\n ingr...	48	292	1,4 mil
13	Guillermo Lasso	@LassoGuillermo	2021-05-24T17:38:49.000Z	Gracias al pueblo ecuatoriano por confiar en m...	784	2,6 mil	16,1 mil
14	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:51:49.000Z	Aspiro a que mis acciones hablen tan elocuent...	1,1 mil	3,8 mil	23,7 mil
15	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:37:51.000Z	Hago un llamado a la UNIDAD NACIONAL, que debe...	227	1,3 mil	5,7 mil
16	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:29:53.000Z	El principio es "no dejar a nadie atrás", lo q...	77	584	2,6 mil
17	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:12:31.000Z	Las causas del #GobiernoDelEncuentro serán las...	83	538	2,1 mil
18	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:09:01.000Z	Que esta democracia que hoy recobramos sea par...	156	1,1 mil	4,9 mil
19	Guillermo Lasso	@LassoGuillermo	2021-05-24T15:59:06.000Z	Empleza el cambio, bienvenidos al Ecuador del ...	1,1 mil	5,7 mil	26,4 mil



Simulación

Tema: Simulación de Eventos Examen.

Ahora procedemos a hacer una limpieza de datos, ya que como podemos ver tenemos datos mezclados con números y letras, lo cual nos dará una complejidad al momento de generar el modelo y la predicción; además realizo cambio en nombre de las columnas para tener una mejor visibilidad y comprensión de cada uno de los datos obtenidos.

```
In [6]: df_tweetsLasso.Comments = df_tweetsLasso.Comments.replace([r'[ \t]+'], ['00'], regex=True)
df_tweetsLasso.Likes = df_tweetsLasso.Likes.replace([r'[ \t]+'], ['00'], regex=True)
df_tweetsLasso.Retweets = df_tweetsLasso.Retweets.replace([r'[ \t]+'], ['00'], regex=True)
df_tweetsLasso.Text = df_tweetsLasso.Text.replace([r'Citar Tweet'], ['Citar Tweet'], regex=True)

df_tweetsLasso['NumHashtags'] = df_tweetsLasso.Text.str.findall(r'#.*(?=\s|$)').str.len()
df_tweetsLasso['WordCount'] = df_tweetsLasso.Text.apply(lambda x: len(x.split()))
df_tweetsLasso['NumLinks'] = df_tweetsLasso.Text.apply(lambda x: 1 if str(x).find('Citar Tweet') != -1 else 0)

df_tweetsLasso = df_tweetsLasso.astype({'Comments': int, 'Likes': int, 'Retweets': int, 'NumHashtags': int, 'WordCount': int, 'NumLinks': int})
df_tweetsLasso.head(20)
```

Out[6]:

	UserName	Handle	Timestamp	Text	Comments	Likes	Retweets	NumHashtags	WordCount	NumLinks
0	Guillermo Lasso	@LassoGuillermo	2021-05-29T21:35:01.000Z	¡Felicidades \n@MorejonGlenda!\nElla y más de...	24	153	762	4	69	1
1	Guillermo Lasso	@LassoGuillermo	2021-05-28T23:57:43.000Z	¡Una buena noticia!\n@EmbajadaChinaEcn me inf...	628	1600	10500	2	34	0
2	Guillermo Lasso	@LassoGuillermo	2021-05-28T22:23:22.000Z	¡Primera visita a Guayas! Nos reunimos en la G...	134	398	200	1	33	0
3	Guillermo Lasso	@LassoGuillermo	2021-05-28T16:59:31.000Z	Los invito a acompañarnos en este emotivo even...	108	293	1100	4	73	1
4	Guillermo Lasso	@LassoGuillermo	2021-05-28T15:37:41.000Z	FAMILIA.\n\n#JuntosLoLogramos	703	1400	16700	1	2	0
5	Guillermo Lasso	@LassoGuillermo	2021-05-28T13:40:49.000Z	Nuestro proyecto de Ley Orgánica de Libre Expr...	268	599	200	2	28	0
6	Guillermo Lasso	@LassoGuillermo	2021-05-28T04:32:57.000Z	Hoy envié a la Asamblea Nacional el proyecto d...	498	100	500	2	35	0
7	Presidencia ECU	@Presidencia_Ec	2021-05-25T21:48:41.000Z	El presidente \n@LassoGuillermo\n, durante su ...	167	558	4500	1	35	0
8	Guillermo Lasso	@LassoGuillermo	2021-05-25T20:04:59.000Z	Hoy tuve la oportunidad de dialogar con \n@leo...	580	878	4300	2	35	0
9	Guillermo Lasso	@LassoGuillermo	2021-05-25T16:16:52.000Z	Fue un honor recibir a representantes de \n@US...	207	638	2900	2	38	0
10	Guillermo Lasso	@LassoGuillermo	2021-05-25T03:11:40.000Z	Esta noche mantuvimos la primera reunión del C...	368	1100	5800	1	40	0
11	Guillermo Lasso	@LassoGuillermo	2021-05-24T19:11:38.000Z	Tengo el corazón lleno de gratitud, cariño y c...	210	966	5300	2	36	0
12	Comunicación Ecuador	@ComunicacionEc	2021-05-24T18:39:24.000Z	AHORITA El presidente \n@LassoGuillermo\n ingr...	48	292	1400	2	49	0
13	Guillermo Lasso	@LassoGuillermo	2021-05-24T17:38:49.000Z	Gracias al pueblo ecuatoriano por confiar en m...	784	2600	16100	2	23	0
14	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:51:49.000Z	Aspiro a que mis acciones hablen tan elocuente...	1100	3800	23700	0	45	0
15	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:37:51.000Z	Hago un llamado a la UNIDAD NACIONAL, que debe...	227	1300	5700	1	43	0
16	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:29:53.000Z	El principio es "no dejar a nadie atrás", lo q...	77	584	2800	2	26	0
17	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:12:31.000Z	Las causas del #GobiernoDelEncuentro serán las...	83	538	2100	2	40	0
18	Guillermo Lasso	@LassoGuillermo	2021-05-24T16:09:01.000Z	Que esta democracia que hoy reobramos sea par...	156	1100	4900	2	36	0
19	Guillermo Lasso	@LassoGuillermo	2021-05-24T15:59:09.000Z	Empezará el cambio, bienvenidos al Ecuador del ...	1100	5700	26400	2	12	0

3. Modelo de regresión Lineal

Para el modelo generamos un entrenamiento en el cual tomamos en cuenta lo que contiene un tweet contra los retweets generados, para ello usaremos un train_test en donde estarán los datos y por medio de una librería (linearRegression) generar un valor de predicción.

```
x = df_tweetsLasso[['Comments', 'Likes', 'NumHashtags', 'WordCount', 'NumLinks']]
y = df_tweetsLasso['Retweets'].values.reshape(-1,1)
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)
regressor = LinearRegression()
regressor.fit(X_train, y_train)

LinearRegression()
```

Ahora ya generado nuestro entrenamiento vamos a generar un score para ver nuestro nivel de predicción.



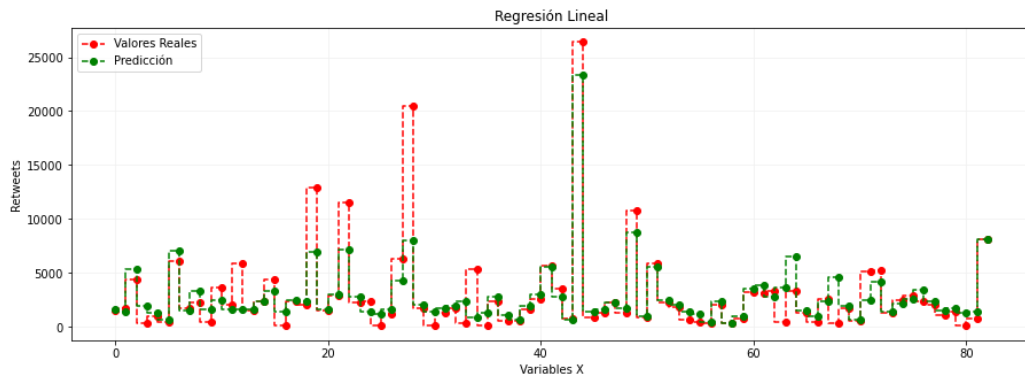
Simulación

Tema: Simulación de Eventos Examen.

```
y_pred = regressor.predict(X_test)
df_tweetsLasso1 = pd.DataFrame({'Data Actual': y_test.flatten().astype(float), 'Data de Predicción': y_pred.flatten()})
df_tweetsLasso1
preRF=r2_score(y_test,y_pred)
print("Precisión del modelo: ", preRF)
```

Precisión del modelo: 0.4439406125453286

Ahora vemos nuestra grafica resultante.



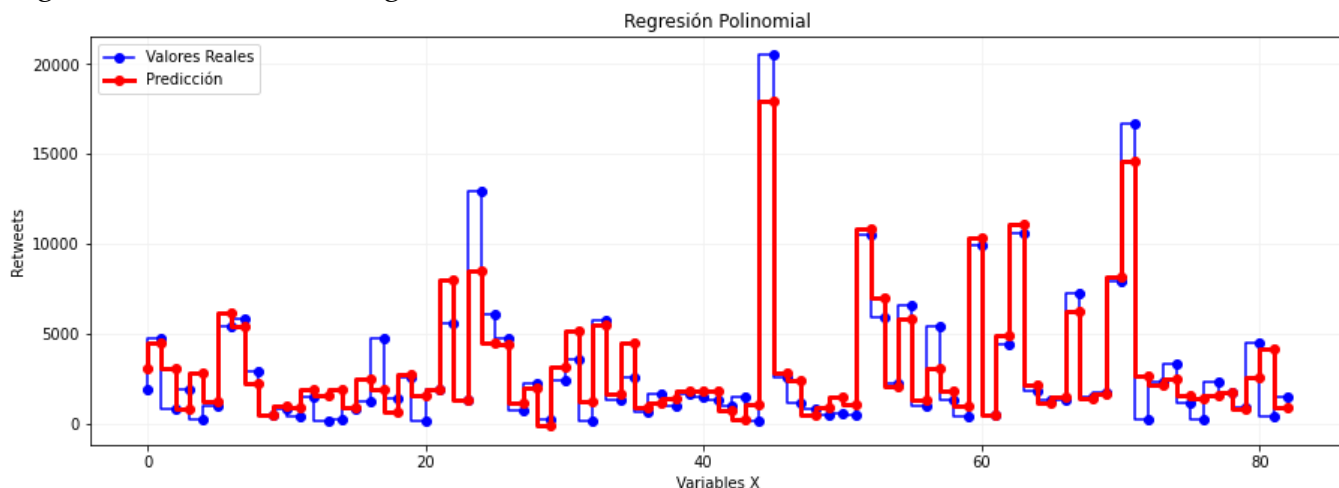
4. Modelo de regresión Polinomial

De igual manera seguimos los pasos anteriores por medio de un entramiento, la única diferencia es que trabajaremos con la librería PolynomialFeatures para nuestra regresión polinomial, dando como resultado de nuestro score

```
y_pred2 = pol_reg.predict(poly_reg.fit_transform(X_test))
df_tweetsLasso2 = pd.DataFrame({'Data Actual': y_test.flatten(), 'Data de Predicción': y_pred2.flatten()})
df_tweetsLasso2
preRF=r2_score(y_test,y_pred2)
print("Precisión del modelo: ", preRF)
```

Precisión del modelo: 0.8873203833953147

La grafica resultante es la siguiente



5. Predicción de re-tweets

Seleccionamos un tweet aleatorio a predecir sus posibles retweets, para ellos trabajaremos con un modelo de regresión lineal y polinomial.

El Tweet seleccionado es sobre la prioridad de vacunación.

	UserName	Handle	Timestamp	Text	Comments	Likes	Retweets	NumHashtags	WordCount	NumLinks
50	Guillermo Lasso	@LassoGuillermo	2021-05-12T16:37:17.000Z	La vacunación es prioridad. Vamos a firmar un ...	842.0	1700.0	10900.0	0.0	44.0	0.0



Simulación

Tema: Simulación de Eventos Examen.

Ahora generamos nuestro entrenamiento y trabajamos con las librerías mencionadas anteriormente; esto nos da como resultado:

```
xTest = df_tweetsLasso[50:51][['Comments', 'Likes', 'NumHashtags', 'WordCount', 'NumLinks']]
print('Predicción regresión lineal:', regressor.predict(xTest))
print('Predicción regresión polinomial: ', pol_reg.predict(poly_reg.fit_transform(xTest)))

Predicción regresión lineal: [[8475.54297364]]
Predicción regresión polinomial: [[9351.60163631]]
```

- Posteriormente se debe seguir un proceso de vacunación en los recintos electorales que se describe a continuación (**Tomar el proceso para el proyecto final**):
 - Solo se va a tener en cuenta uno de los recintos electorales (investigar datos de cuantas personas asisten a votar).
 - Tomar los resultados de la regresión para la vacuna según la llegada.
 - Se tiene un promedio que el 80% de personas realizarán el proceso de vacunación dentro del Ecuador.
 - Dentro del proceso se tiene que alrededor del 5% - 10% no podrán vacunarse.
 - Las personas solo tienen un recinto electoral para realizar el proceso.
 - Las personas realizan la primera vacuna y 30 días después la segunda vacuna.
 - La persona se acerca a la mesa y hacen fila en caso de ser necesario para recibir la vacuna.
 - Realiza la vacunación en un tiempo aleatorio entre 5 a 10 minutos.
 - Debe esperar 20 minutos dentro del establecimiento para verificar que no tenga problemas de salud.
 - La persona recibe su certificado de vacunación y la fecha de la próxima vacuna entre 2 – 3 minutos.
 - La persona sale del recinto electoral.
 - Regresan para la próxima fecha y se repite el ciclo.
- El proceso de simulación desarrollado deberá considerar los siguientes aspectos:
 - Se debe establecer un modelo basado en modelos matemáticos para la predicción del número de veces que se compartirá o la tendencia del presidente basada en redes sociales.
 - El programa deberá generar gráficas que indiquen la ecuación matemática de las tendencias.
 - Deben calcularse las siguientes métricas del sistema de simulación de eventos discretos:
 - Total de personas que realizaron el proceso de vacunación.
 - Gráfico del porcentaje de personas que no recibieron la vacuna.
 - El tiempo promedio de espera.



DESARROLLO PROCESO DE VACUNACIÓN EN LOS RECINTOS ELECTORALES (PUNTO 2)

1. El recinto escogido fue el del Colegio Maria Auxiliadora de la parroquia el sagrario, ya que es mi lugar de votación, en donde se registro 7286 personas empadronadas en este recinto.

Establecemos datos y variables para nuestras funciones en donde estableceremos los tiempos para cada proceso que realiza para vacunación.

```
# Votantes en el COLEGIO MARIA AUXILIADORA - PARROQUIA EL SAGRARIO
TOTAL_PERSONAS = 7.286
#Mesa de vacunacion
MESA_VACUNACION = 20
# Intervalo de tiempo de control para que la persona se registre "minutos"
INGRESO_MESA = 5
# Tiempo para vacunar
VACUNAR = 6
# Verificacion de salud post vacuna
VERIFICACION_POST = 20
# Tiempo de simulación 2H
TIEMPO_SIMULACION = 120
# Vacunados
TOTAL_VACUNADOS = 0
# Cantidad de personas que Llegaron
TOTAL_PERSONAS_COLEGIO = 0
# Tiempo Vacunación
TIEMPO_VACUNA = 0
# Tiempo Espera
TIEMPO_ESPERA = 0
# Tiempo de vacunación de cada persona
TIEMPO_VACUNA_ARR = {}
# Tiempo de espera de cada persona
TIEMPO_ESPERA_ARR = {}
```

2. Generamos una clase en donde estarán cada uno de nuestros métodos para generar y obtener los tiempos.

Como primer punto tomamos en cuenta las mesas que existen ya que con ello podemos determinar el tiempo por persona vacunada desde que ingresa al recinto, posterior a ello lo que se realiza es una verificación de salud, esto se refiere a que una vez vacunada la persona debe tomar un tiempo de descanso para que no tenga efectos secundarios, o en consecuencia si tiene el tiempo que toma para que un medico lo entienda; una vez culminado la vacunación se medirá el tiempo que toma en entregar el certificado, estos procesos son secuenciales.

```
def __init__(self, environment, mesa_vacunacion, esperar_turno, chequeo_post):
    self.env = environment
    # Creamos el recurso que representa las mesas para la vacunacion de las electores
    self.mesa_vacunacion = simpy.Resource(environment, mesa_vacunacion)
    # Variables
    self.esperar_turno = esperar_turno
    self.chequeo_post = chequeo_post

def vacunar_persona(self, p): #Funcion para vacunar
    yield self.env.timeout(rm.randint(5,10))

def verificacion_salud(self):
    yield self.env.timeout(self.chequeo_post)

def recibir_certificado(self):
    yield self.env.timeout(rm.randint(2,3))

def esperar(self):
    yield self.env.timeout(self.esperar_turno)
```




Simulación

Tema: Simulación de Eventos Examen.

3. Métodos Usados

La medición de tiempos se realizarán según van ingresado a la vacunación, además el tiempo que toma en espera de la fila, se ira incrementando cada método una vez que la persona antes haya pasado.

```
def p_vacunados():
    global TOTAL_VACUNADOS
    TOTAL_VACUNADOS +=1

def personas_llegaron():
    global TOTAL_PERSONAS_COLEGIO
    TOTAL_PERSONAS_COLEGIO +=1

def espera(persona, espera):
    global TIEMPO_ESPERA
    global TIEMPO_ESPERA_ARR

    TIEMPO_ESPERA_ARR[persona.split('-')[0]] = espera
    TIEMPO_ESPERA += espera

def vacunar(persona, tiempoPersona):
    global TIEMPO_TOTAL_VACUNACION
    global TIEMPO_VACUNA_ARR

    TIEMPO_VACUNA_ARR[persona.split('-')[0]] = tiempoPersona
    TIEMPO_TOTAL_VACUNACION += tiempoPersona

def entrada_recinto(env, p_identificacion, vacunacion):

    yield env.process(vacunacion.esperar())
    start_espera = env.now
```

Ahora para determinar los tiempos por persona dentro del recinto se realiza lo siguiente: la persona que ingresa se le da un tiempo determinado, en el cual inicia el proceso de vacunación entregando su identificación, posterior a ello se le pone la vacuna dado como resultado un tiempo extra, ahora se valida el tiempo en el cual se le hace una valoración medica, una vez termina este proceso llamamos a la función que nos permite generar el tiempo que se valida que no exista efectos secundarios; ahora guardamos esos tiempos y los vamos sumando y almacenando para dar un resultado en tiempo y fechas próximas para la vacunación, además se genera el tiempo de la entrega del certificado.

```
def entrada_recinto(env, p_identificacion, vacunacion):

    yield env.process(vacunacion.esperar())
    start_espera = env.now

    rand_vacuna = rm.randrange(1, 100)
    if rand_vacuna <= 10:
        print('La persona con el uid: %s, no puede ser vacunada' % (p_identificacion))
    else:
        # Especificamos que vamos a usar un recurso (Resource) que representa una mesa
        with vacunacion.mesa_vacunacion.request() as mesa:

            ##INICIO
            print('La persona %s, ingresa a vacunarse, hora de ingreso: %.2f' % (p_identificacion, env.now))
            yield mesa

            ##INICIO TIEMPO
            inicio = env.now

            yield env.process(vacunacion.vacunar_persona(p_identificacion)) #Se inicia el proceso de vacunacion
            p_vacunados()

            ##FIN TIEMPO
            end_vacunacion = env.now - inicio

            vacunar(p_identificacion, end_vacunacion)
            print('La Persona %s fue vacunada, hora vacunacion: %.2f' % (p_identificacion, env.now))

            yield env.process(vacunacion.verificacion_salud())
            print('Ingreso chequeo medico, persona %s, hora de ingreso: %.2f' % (p_identificacion, env.now))

            ##CERTIFICADO

            segunda_dosis = date.today() + datetime.timedelta(days=30)
            espera(p_identificacion, env.now - inicio)

            print('La persona %s recibe su certificado, fecha segunda dosis: %s' % (p_identificacion, segunda_dosis))
```




Simulación

Tema: Simulación de Eventos Examen.

Este proceso nos genera la siguiente información (cabe recalcar que se toma en cuenta que un 5%-10% no podrá vacunarse dependiendo de su sistema y/o enfermedades)

Vacunacion Colegio Maxia Auxiliadora

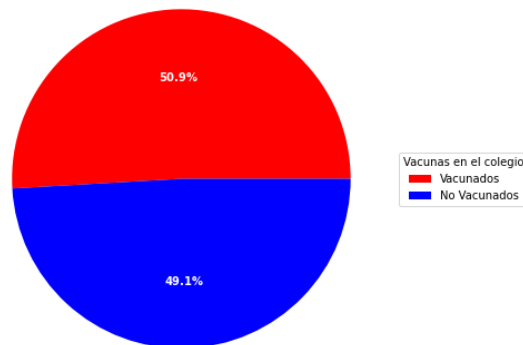
```
La persona 1, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 2, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 3, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 4, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 5, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 6, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 7, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 8, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 9, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona con el uid: 10, no puede ser vacunada  
La persona 11, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 12, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 13, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 14, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona con el uid: 15, no puede ser vacunada  
La persona 16, ingresa a vacunarse, hora de ingreso: 5.00.  
La persona 17, ingresa a vacunarse, hora de ingreso: 5.00.
```

Dando como resultado un total de 84 personas que cumplieron todo el proceso sin complicaciones y un promedio de tiempo de espera de 40,19min

```
38]: #Total de personas vacunadas:  
print(TOTAL_VACUNADOS)  
  
84  
  
39]: #Promedio espera:  
promedio = list(map(int, TIEMPO_ESPERA_ARR))  
for i in promedio:  
    a = sum(promedio)  
  
promedio_total = a/TOTAL_VACUNADOS  
print(promedio_total)  
  
40.19047619047619
```

4. Grafica personas vacunadas vs no vacunadas

VACUNACION COVID 19





Simulación

Tema: Simulación de Eventos Examen.



- **Fecha de presentación: 01/06/2021 – 23:55.**
- El informe en pdf (Archivo Notebook) se deberá subir al Git personal, todo el sistema debe ser basado en Python – Sympy - Open Source.

Opcional: Cualquier adición o librerías que no se revisaron en clases serán tomada como puntos adicionales al interciclo.

Proyecto Final Presentación: 08/06/2021

Referencias:

- [1]: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6291769/>
- [2]: <https://www.aprendemachinelearning.com/regresion-lineal-en-espanol-con-python/>
- [3]: <https://eprints.ucm.es/48804/1/TFM%20Manuel%20Alejandro%20Rodriguez%20Santana.pdf>
- [4]: http://opac.pucv.cl/pucv_txt/txt-8000/UCC8094_01.pdf