



## Simulación

Tema: Simulación de Eventos.

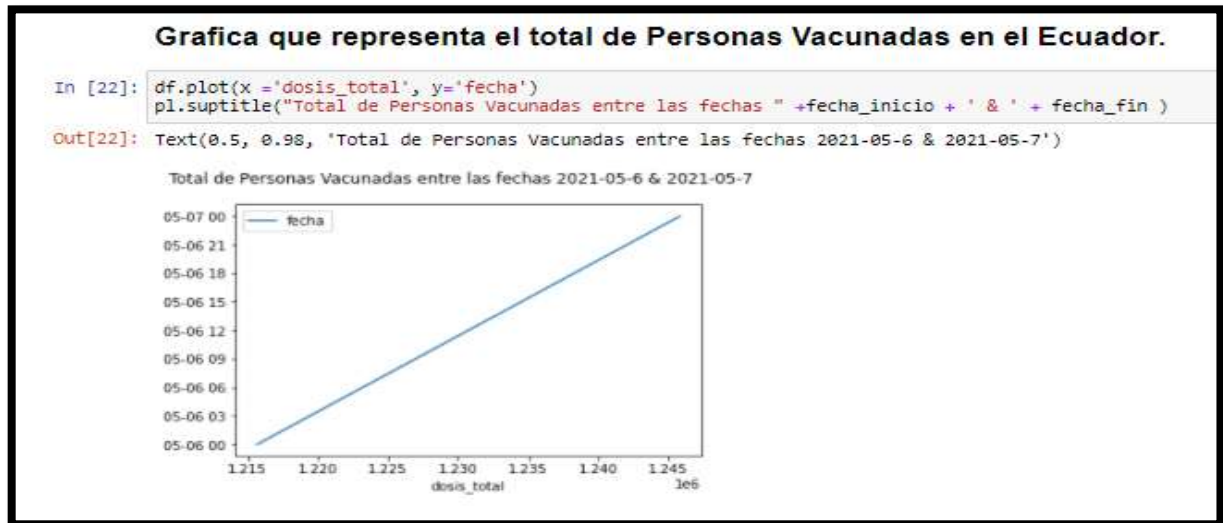
### PRUEBA PRACTICA SIMULACIÓN

FECHA: 12/05/2021

NOMBRE: VÁSQUEZ FAJARDO FRANKLIN JOEL

## 1. Generar graficas para entender y procesar los datos:

### 1.1. Total, de personas Vacunadas



### 1.2. Fabricantes de la vacuna

#### Grafica de pie que representa los fabricantes de Vacunas.

```
df_fabricantes = pd.read_csv('C:/Users/fvasq/Desktop/UPS/10 Ciclo/Simulacion/in/vacunas/fabricantes.csv')
df_fabricantes['arrived_at'] = pd.to_datetime(df_fabricantes['arrived_at'],format='%d/%m/%Y')
df_copy = df_fabricantes
df_fabricantes.head()
```

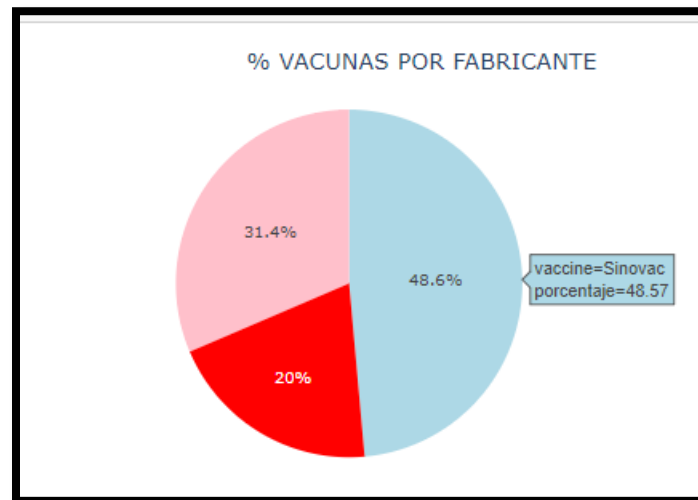
0]:

	vaccine	total	arrived_at
0	Pfizer/BioNTech	8190	2021-01-20
1	Pfizer/BioNTech	16380	2021-02-17
2	Pfizer/BioNTech	17550	2021-02-24
3	Pfizer/BioNTech	31590	2021-03-03
4	Sinovac	20000	2021-03-08



## Simulación

Tema: Simulación de Eventos.



### 1.3. Histograma de vacunas por mes de llegada y fabricante.

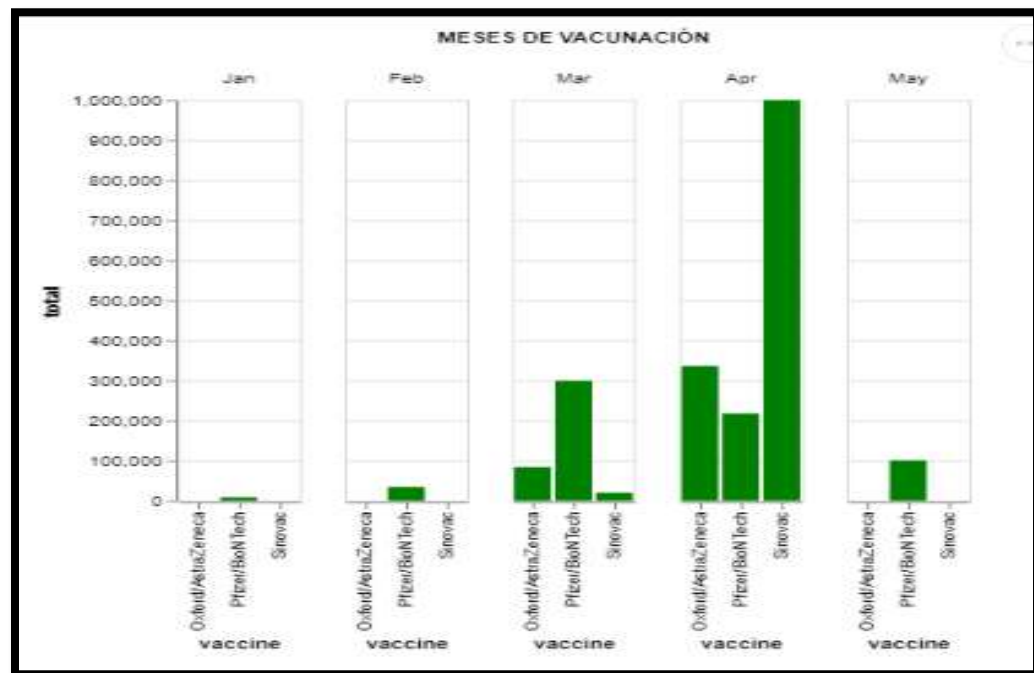
vaccine	arrived_at	
Oxford/AstraZeneca	2021-03-31	84000
	2021-04-30	336000
Pfizer/BioNTech	2021-01-31	8190
	2021-02-28	33930
	2021-03-31	299520
	2021-04-30	217620
	2021-05-31	100620
Sinovac	2021-03-31	20000
	2021-04-30	1000000

```
df_vacunas.reset_index(inplace=True)
histograma= alt.Chart(df_vacunas).mark_bar().encode(
  x=alt.X('vaccine'),
  y=alt.Y('total'),
  tooltip=['vaccine', 'total'],
  color=alt.condition( alt.datum.total == 1200000,alt.value('green'),alt.value('green')),
  column=alt.Column('month(arrived_at)',title='MESES DE VACUNACIÓN'))
histograma
##altair_viewer.show(histograma)
```



## Simulación

Tema: Simulación de Eventos.



2. Generar un reporte parametrizado con las fechas para obtener datos del primer punto.

```
import argparse
import papermill as pm

def main(fechas):
    pm.execute_notebook(
        './entradaFinal.ipynb',
        './out/salidaFinal.ipynb',
        parameters=dict(fecha_inicio=fechas[0], fecha_fin=fechas[1])
    )

if __name__ == '__main__':
    args = argparse.ArgumentParser()

    args.add_argument('date',
                      help='Fecha Inicial y Fecha Final',
                      type=str
    )

    arguments = args.parse_args()
    main([i for i in arguments.date.split(',')])
```

3. Modelo matemático de predicción basado en regresión del proceso de vacunación en base al número de vacunados (1 y 2 Dosis)

En este caso usamos los modelos de regresión lineal y por Random Forest para realizar el proceso de comparación, una vez realizado los modelos pudimos comprobar que el modelo Random Forest nos arrojó mejores resultados obteniendo así una mejor precisión.



## Simulación

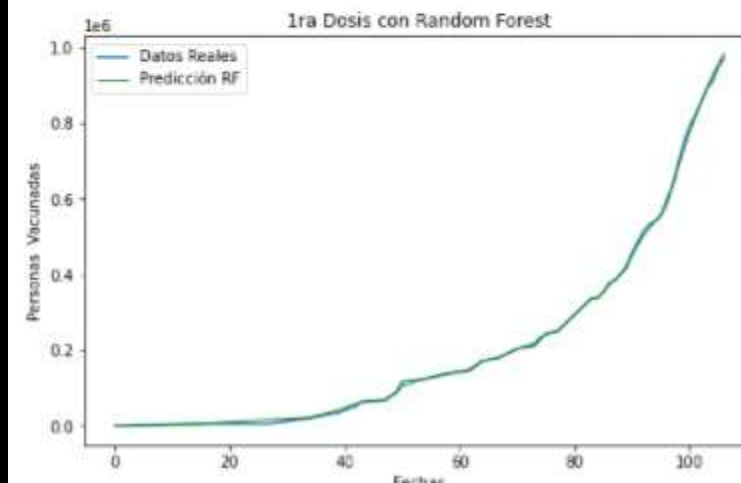
Tema: Simulación de Eventos.

### 3.1. Random Forest Primera y segunda Dosis

```
x = vacunados['fecha'].values.reshape(-1,1)
y = vacunados['primera_dosis'].values.reshape(-1,1)

random = RandomForestRegressor(n_estimators=10, random_state=0)
random.fit(np.array(x).reshape(-1, 1), y.ravel())
prediccion = random.predict(x)

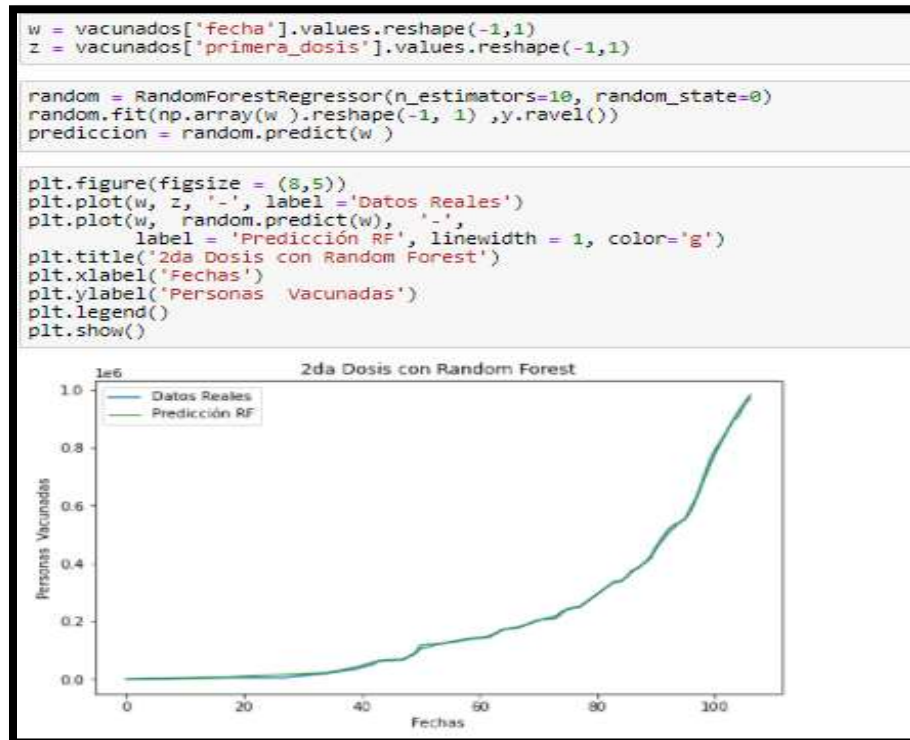
plt.figure(figsize = (8,5))
plt.plot(x, y, '-', label = 'Datos Reales')
plt.plot(x, random.predict(x), '-',
         label = 'Predicción RF', linewidth = 1, color='g')
plt.title('1ra Dosis con Random Forest')
plt.xlabel('Fechas')
plt.ylabel('Personas Vacunadas')
plt.legend()
plt.show()
```



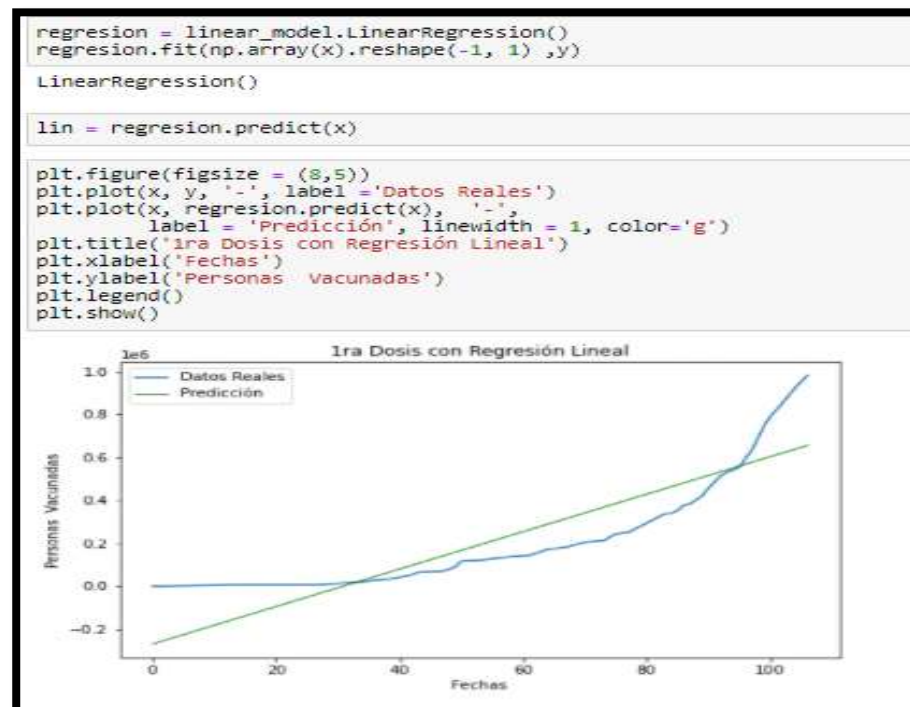


## Simulación

Tema: Simulación de Eventos.



### 3.2. Regresión Lineal Primera y segunda dosis





## Simulación

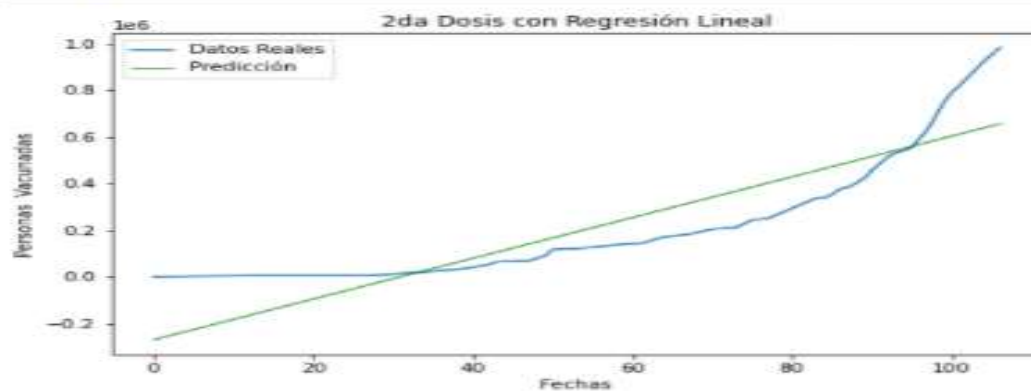
Tema: Simulación de Eventos.

```
regression = linear_model.LinearRegression()
regression.fit(np.array(w).reshape(-1, 1), z)

LinearRegression()

lin = regression.predict(w)

plt.figure(figsize = (8,5))
plt.plot(w, z, '-', label='Datos Reales')
plt.plot(w, regression.predict(w), '-',
        label = 'Predicción', linewidth = 1, color='g')
plt.title('2da Dosis con Regresión Lineal')
plt.xlabel('Fechas')
plt.ylabel('Personas Vacunadas')
plt.legend()
plt.show()
```





## Simulación

### Tema: Simulación de Eventos.

#### 4. Proceso de comparación con al menos dos países.

##### 4.1. Comparación con primera y segunda dosis

### INDIA

```
: india = pd.read_csv('C:/Users/fvasq/Desktop/UPS/10 Ciclo/Simulacion/in/vacunas/India.csv')
india['date'] = pd.to_datetime(india['date'],format='%Y-%m-%d')
fecha = india['date']
india['date'] = fecha.map(lambda x : (x - primera).days)
```

```
: india.head()
```

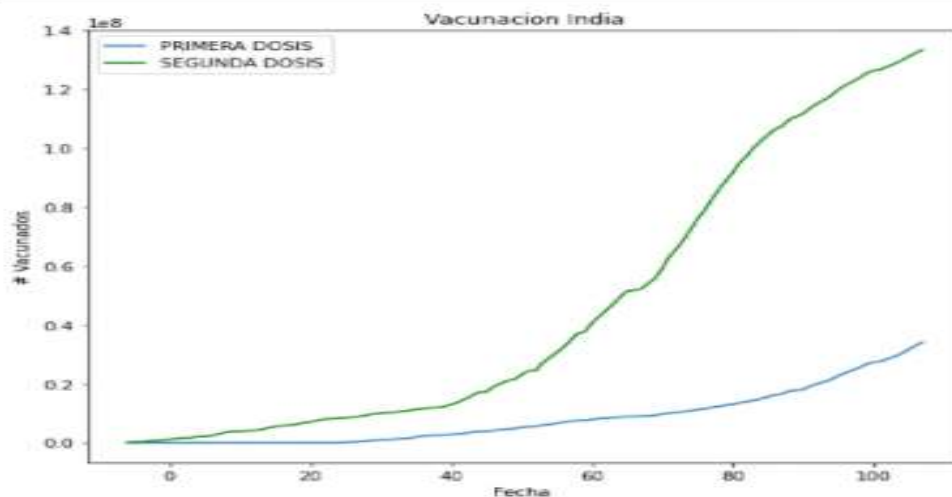
	location	date	vaccine	source_url	total_vaccinations	people_vaccinated	people_fully_vaccinated
0	India	-8	Covaxin, Oxford/AstraZeneca	<a href="https://twitter.com/MoHFW_INDIA/status/1350459...">https://twitter.com/MoHFW_INDIA/status/1350459...</a>	0	0	0
1	India	-5	Covaxin, Oxford/AstraZeneca	<a href="https://twitter.com/MoHFW_INDIA/status/1350459...">https://twitter.com/MoHFW_INDIA/status/1350459...</a>	191181	191181	0
2	India	-4	Covaxin, Oxford/AstraZeneca	<a href="https://twitter.com/MoHFW_INDIA/status/1350815...">https://twitter.com/MoHFW_INDIA/status/1350815...</a>	224301	224301	0
3	India	-3	Covaxin, Oxford/AstraZeneca	<a href="https://www.mohfw.gov.in/">https://www.mohfw.gov.in/</a>	454049	454049	0
4	India	-2	Covaxin, Oxford/AstraZeneca	<a href="https://www.mohfw.gov.in/">https://www.mohfw.gov.in/</a>	674835	674835	0

```
: box = india['date'].values.reshape(-1,1)
boy = india['people_vaccinated'].values.reshape(-1,1)
randomin = RandomForestRegressor(n_estimators=10, random_state=0)
```

```
bolw = india['date'].values.reshape(-1,1)
bolz = india['people_fully_vaccinated'].values.reshape(-1,1)
randomin2 = RandomForestRegressor(n_estimators=10, random_state=0)
randomin2.fit(bolw, bolz.ravel())
```

```
RandomForestRegressor(n_estimators=10, random_state=0)
```

```
plt.figure(figsize = (8,7))
plt.plot(bolw, bolz, '-', label = 'PRIMERA DOSIS')
plt.plot(box, boy, '-', label = 'SEGUNDA DOSIS', color='g')
plt.title('Vacunacion India')
plt.xlabel('Fecha ')
plt.ylabel('# Vacunados')
plt.legend()
plt.show()
```







## Simulación

### Tema: Simulación de Eventos.

#### ARUBA

```
argent = pd.read_csv('C:/Users/fvasq/Desktop/UPS/10 Ciclo/Simulacion/in/vacunas/aruba.csv')
argent['date'] = pd.to_datetime(argent['date'], format='%Y-%m-%d')
fecha = argent['date']
argent['date'] = fecha.map(lambda x : (x - primera).days)
```

```
argent.head()
```

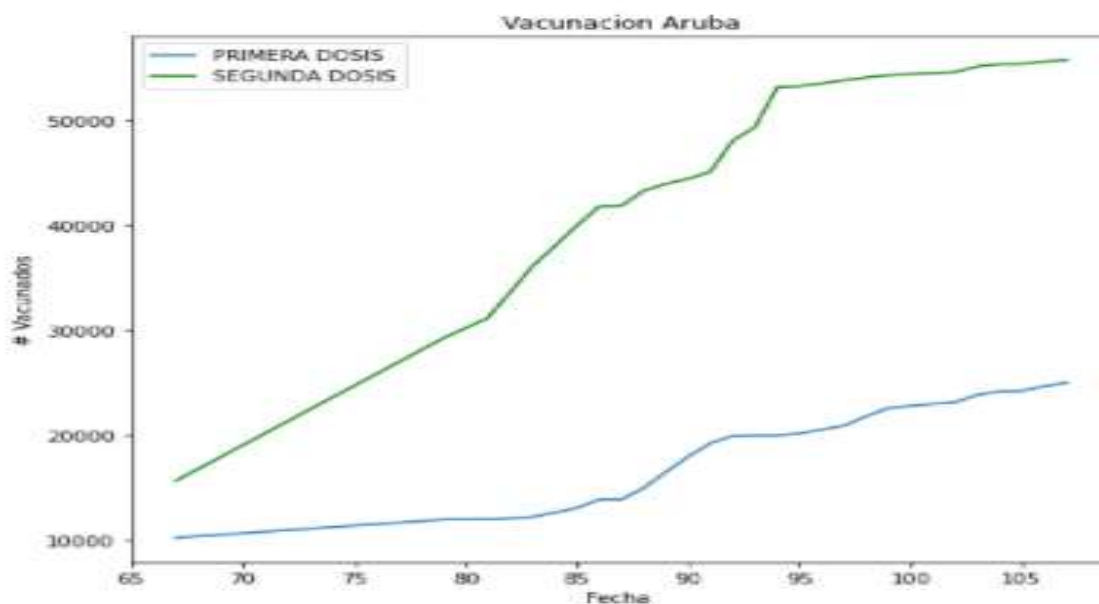
	location	date	vaccine	source_url	total_vaccinations	people_vaccinated	people_fully_vaccinated
0	Aruba	67	Pfizer/BioNTech	<a href="https://www.facebook.com/desaruba/photos/4207...">https://www.facebook.com/desaruba/photos/4207...</a>	25768	15600	10168
1	Aruba	79	Pfizer/BioNTech	<a href="https://www.government.aw">https://www.government.aw</a>	41098	29197	11899
2	Aruba	81	Pfizer/BioNTech	<a href="https://www.government.aw">https://www.government.aw</a>	43073	31139	11934
3	Aruba	82	Pfizer/BioNTech	<a href="https://www.government.aw">https://www.government.aw</a>	45588	33563	12025
4	Aruba	83	Pfizer/BioNTech	<a href="https://www.government.aw">https://www.government.aw</a>	48200	36058	12142

```
arx = argent['date'].values.reshape(-1,1)
ary = argent['people_vaccinated'].values.reshape(-1,1)
randomAr = RandomForestRegressor(n_estimators=10, random_state=0)
randomAr.fit(arx, ary.ravel())
```

```
arw = argent['date'].values.reshape(-1,1)
arz = argent['people_fully_vaccinated'].values.reshape(-1,1)
randomAr = RandomForestRegressor(n_estimators=10, random_state=0)
randomAr.fit(arw, arz.ravel())
```

```
RandomForestRegressor(n_estimators=10, random_state=0)
```

```
plt.figure(figsize = (8,7))
plt.plot(arw, arz, '-', label = 'PRIMERA DOSIS')
plt.plot(arx, ary, '-', label = 'SEGUNDA DOSIS', color='g')
plt.title('Vacunacion Aruba')
plt.xlabel('Fecha ')
plt.ylabel('# Vacunados')
plt.legend()
plt.show()
```





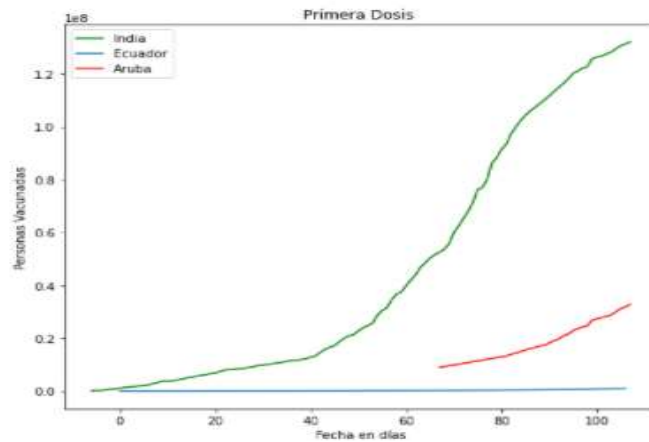


## Simulación

Tema: Simulación de Eventos.

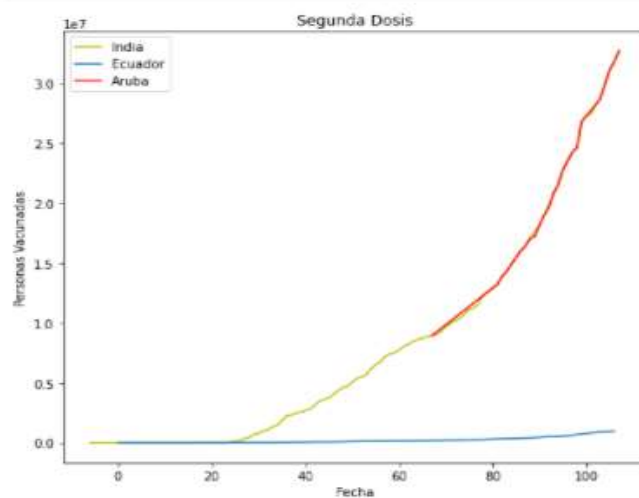
### APLICACION PRIMERA DOSIS

```
In [416]: plt.figure(figsize = (8,7))
plt.plot(box, randomin.predict(box), '-', label = 'India', color='g')
plt.plot(x, random.predict(x), '-', label = 'Ecuador')
plt.plot(arx, randomAr.predict(arx), '-', label = 'Aruba', color='r')
plt.title('Primera Dosis')
plt.xlabel('Fecha en días')
plt.ylabel('Personas Vacunadas')
plt.legend()
plt.show()
```



### APLICACION SEGUNDA DOSIS

```
In [417]: plt.figure(figsize = (8,7))
plt.plot(bolw, randomin2.predict(bolw), '-', label = 'India', color='y')
plt.plot(w, random.predict(w), '-', label = 'Ecuador')
plt.plot(arw, randomAr.predict(arw), '-', label = 'Aruba', color='r')
plt.title('Segunda Dosis')
plt.xlabel('Fecha')
plt.ylabel('Personas Vacunadas')
plt.legend()
plt.show()
```





## Simulación

Tema: Simulación de Eventos.

### 5. Fecha tentativa en la que todos los ecuatorianos estarán vacunados.

#### Predicción Vacunas Total

```
1] tentativa = max(w)+1
prediccion = regresion.predict([tentativa])
datos = [17370000]
while prediccion <= datos:
    tentativa = tentativa +1
    prediccion = regresion.predict([tentativa])

fecha = primera + datetime.timedelta(days=np.float64(tentativa[0]))
fecha
2]: Timestamp('2036-11-20 00:00:00')
```

### 6. Opinión

Después de realizar nuestros modelos de regresión y basándonos en datos reales de vacunación a nivel mundial, pudimos constatar el retraso y poca organización que se tiene con respecto al tema de vacunación dentro de nuestro país, ya que se pudo verificar que el tema de vacunación en el Ecuador avanza de una manera demasiado lenta y si seguimos a este ritmo nos tardaremos demasiado en vacunar a todos los ecuatorianos. En los países contra los que se realizó una comparación se puede observar que en poco tiempo más estarán vacunados el total de la población a comparación de Ecuador donde llevara demasiado tiempo completar el proceso.

### 7. Conclusiones

- Pudimos constatar que con nuestro modelo de regresión Random Forest obtuvimos una mayor precisión que, con nuestro modelo de regresión lineal, por lo cual al momento de realizar nuestras comparaciones con otros países utilizamos dicho modelo para tener una mejor precisión.

### 8. Recomendaciones

- Se recomienda al momento de realizar los modelos de regresión, hacerlo con datos de vacunas y de países actualizados para así ver cómo va avanzando el proceso de forma actualizada.

### 9. Referencias

[1]: <https://www.salud.gob.ec/plan-nacional-de-vacunacion-ecuador-2021-llega-segundo-lote-de-vacunas-pfizer-por-16-380-dosis/>

[2]: <https://www.edicionmedica.ec/secciones/salud-publica/-cuantas-personas-ya-se-han-vacunado-contr-la-covid-19-en-el-ecuador--97224>