

Fluidez Racial en las Elecciones de Brasil

Machine Learning & Data Analysis: Presentación Final

Franco Albino

6 de Julio del 2023

La función a estimar:

$$VS_i = \beta_1 W_i + \beta_2 X_i + \epsilon_i$$

- VS_i : Porcentaje de votos del candidato i .
- W_i : Si el candidato i declaró ser “blanco”.
- X_i : Otras variables relevantes.

... Deberíamos encontrar que $\beta_1 > 0$.

Table 1: Declared race in Brazilian municipal elections

	Race in 2020						N
	White <i>Branco</i>	Mixed <i>Pardo</i>	Black <i>Preto</i>	Asian <i>Amarelo</i>	Indigenous <i>Indígena</i>	Not declared	
Race in 2016							
White	78.68%	19.22%	1.00%	0.28%	0.03%	0.79%	76 316
Mixed	20.67%	68.56%	8.84%	0.32%	0.18%	1.44%	59 548
Black	4.05%	31.48%	63.07%	0.07%	0.12%	1.21%	12 904
Asian	36.86%	44.20%	2.79%	14.24%	0.15%	1.76%	681
Indigenous	3.21%	28.06%	4.61%	0.00%	63.13%	1.00%	499
N	73 139	59 996	14 208	512	462	1 631	149 948
%	48,78	40,01	9.48	0.34	0.30	1.09	100.00

Hipótesis de Comportamiento

Cada candidato tiene que resolver un problema de maximización restringido:

$$\max_{R_i, X_i} VS_i(R_i, X_i)$$

sujeto a una serie de restricciones.

- Una de ellas es el verdadero **tono de piel** de los candidatos...
- Supuesto: Todo lo demás constante, es un disvalor que los candidatos “mientan” con su raza.
- Se deriva que la “función de demanda” de cada raza por parte de cada candidato depende del su tono de piel.

Identificación del Tono de Piel usando ML



Figure 1: Candidatos Elecciones Municipales

El Problema de las Fotos en Blanco y Negro



Figure 2: Muchas de las fotos de la base de datos son en blanco y negro...

Algoritmo para Identificación y Filtrado de Fotos en Blanco Y Negro usando ML

- 1 A mano generar dos carpetas con fotos: una con fotos a color y otra con fotos en blanco y negro. Esto va a servir como sustento para que la máquina aprenda a reconocer de qué tipo es cada foto.
- 2 Generar el código que, a partir de las carpetas de aprendizaje, llegue a la siguiente función:

$$Es\ Blanco\ y\ Negro = f(Foto)$$

- 3 Usar la función para categorizar las fotos en una carpeta con muchas de ellas.
- 4 Output: Dos carpetas. Una con las fotos que la función predice a color y otra con las que predice ser blanco y negro.

Algoritmo CASCo (Classsification Algorithm for Skin Color)

El objetivo es estimar la siguiente función:

$$\textit{Tono de Piel} = f(\textit{Foto})$$

Para eso usamos el algoritmo CASCo:

- 1 Detecta el area de la cara.
- 2 Extrae las areas que no contienen piel.
- 3 Obtiene el color dominante usando algoritmo **k-means**.
- 4 Asigna la foto a una categoría.

Resultados

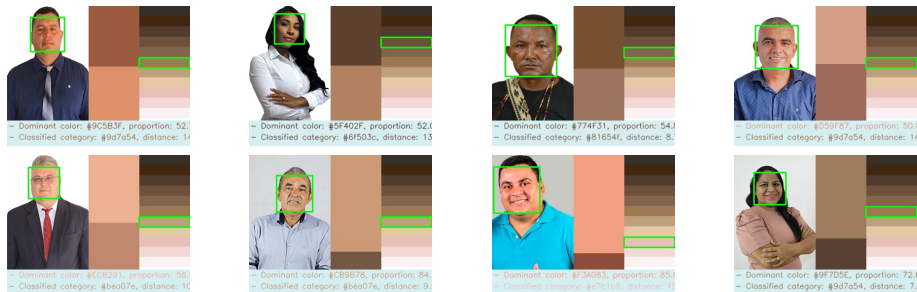
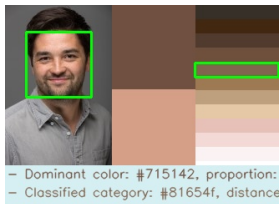
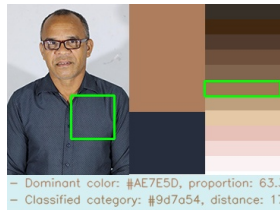


Figure 3: Resultados del algoritmo CASCo

Errores en la clasificación



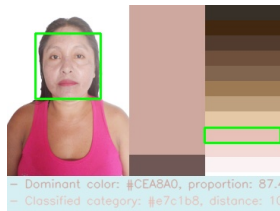
(a) Too dark



(b) Wrong Face ID



(c) No Face ID



(d) Too bright

Cuantificación de los errores

De una muestra aleatoria de 100 fotos se obtuvieron a mano los siguientes resultados:

Error	Ocurrencias	Porcentaje
Black & White	1	1%
No Face ID	6	6%
Wrong Face ID	7	7%
Too Dark	20	20%
Too bright	5	5%

Figure 5: Tipos de errores en muestra aleatoria de 100 fotos

Contraste de Hipótesis: ¿Es el Tono de Piel una Restricción al Momento de Elegir la Raza?

- Para analizar la pregunta mergeamos los resultados a la base de datos del proyecto...
- Y correlacionamos la raza autodeclarada con el tono de piel:

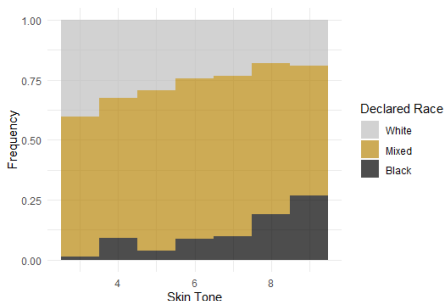


Figure 6: Frecuencia del tono de piel según raza autodeclarada

Conclusiones

- Como esperábamos, la elección de la variable raza por parte del candidato parece estar restringida por su propio tono de piel.
- A pesar de eso, existe cierta volatilidad en los resultados, que se puede interpretar como que los candidatos tienen en realidad bastante espacio para elegir la raza que prefieran.
- Sin embargo, los resultados deben ser tomados con pinza debido a la existencia de errores en los algoritmos que generaron la variable “Tono de Piel”.
- Trabajo futuro: Poder reconocer patrones que nos ayuden a identificar los posibles sesgos en el algoritmo que generó la variable para así poder obtener resultados más confiables.