

¿Podemos
predecir el
éxito
estudiantil
usando
estadística?

```
def __init__(self, path):
    self.file = None
    self.fingerprints = set()
    self.logdups = True
    self.debug = debug
    self.logger = logging.getLogger(__name__)
    if path:
        self.file = open(os.path.join(path, 'requests.json'), 'w')
        self.file.seek(0)
        self.fingerprints.update(self._get_fingerprints())

    @classmethod
    def from_settings(cls, settings):
        debug = settings.getbool('SUPERFILTER_DEBUG')
        return cls(job_dir(settings), debug)

    def request_seen(self, request):
        fp = self.request_fingerprint(request)
        if fp in self.fingerprints:
            return True
        self.fingerprints.add(fp)
        if self.file:
            self.file.write(fp + os.linesep)

    def request_fingerprint(self, request):
        return request_fingerprint(request)
```

Equipo



Juan David
Valencia



David
Cardona



Kevin
Loaiza



Miguel
Correa

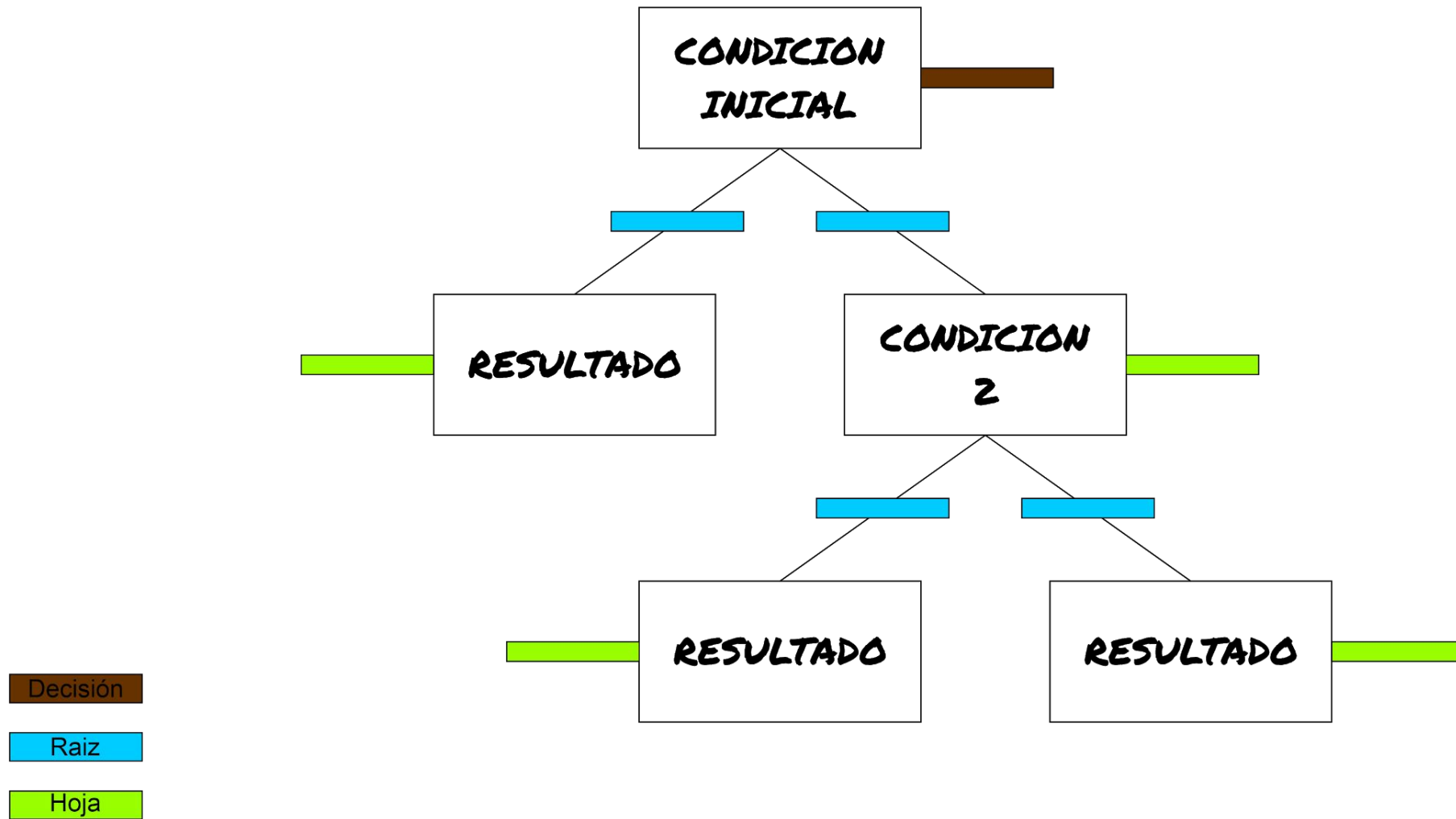


Mauricio
Toro



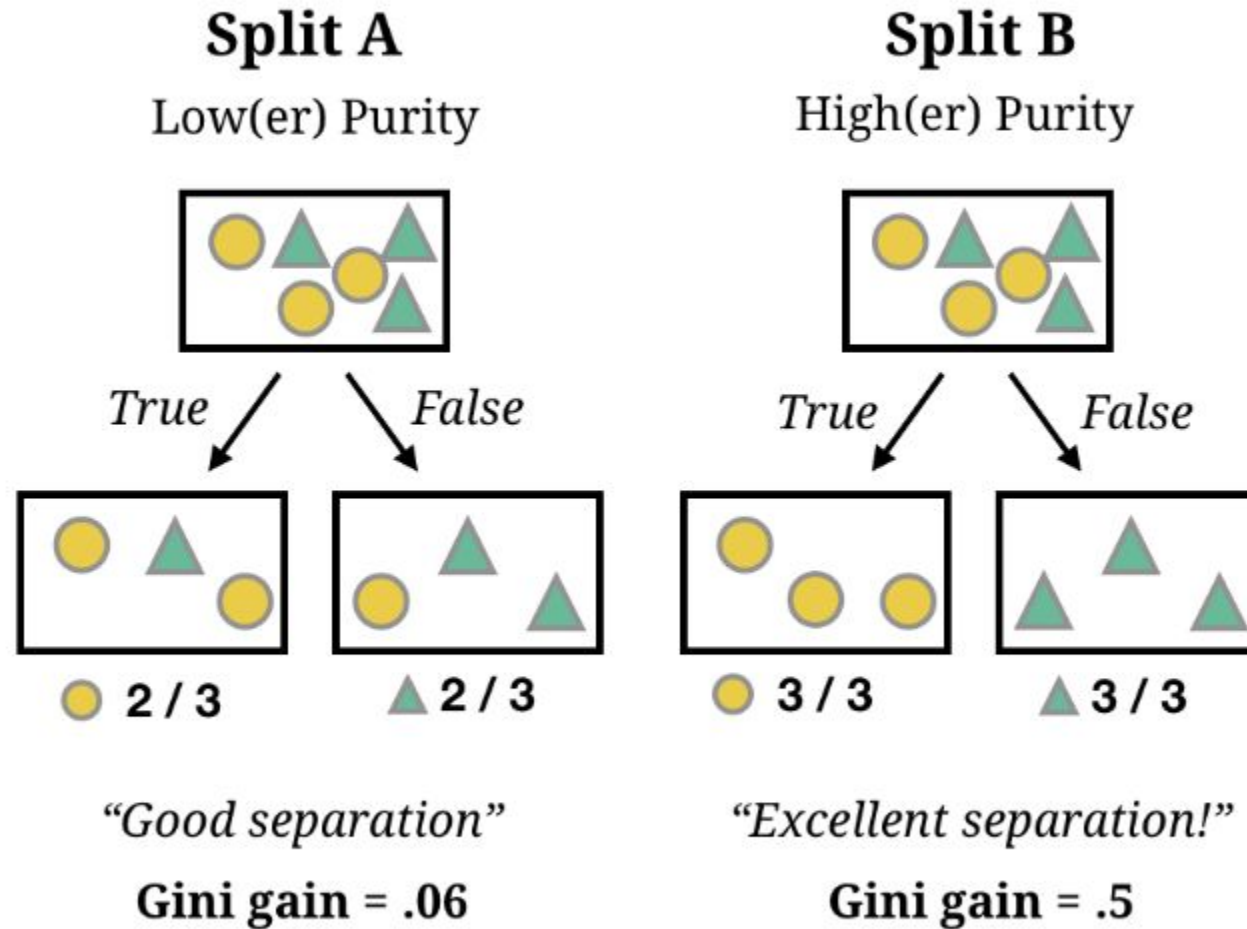
<https://github.com/jdvalencit/ST0245-001/tree/master/proyecto>





Las hojas del árbol se crearán en base al algoritmo ID3 y las impurezas de gini obtenidas al dividir la matriz.

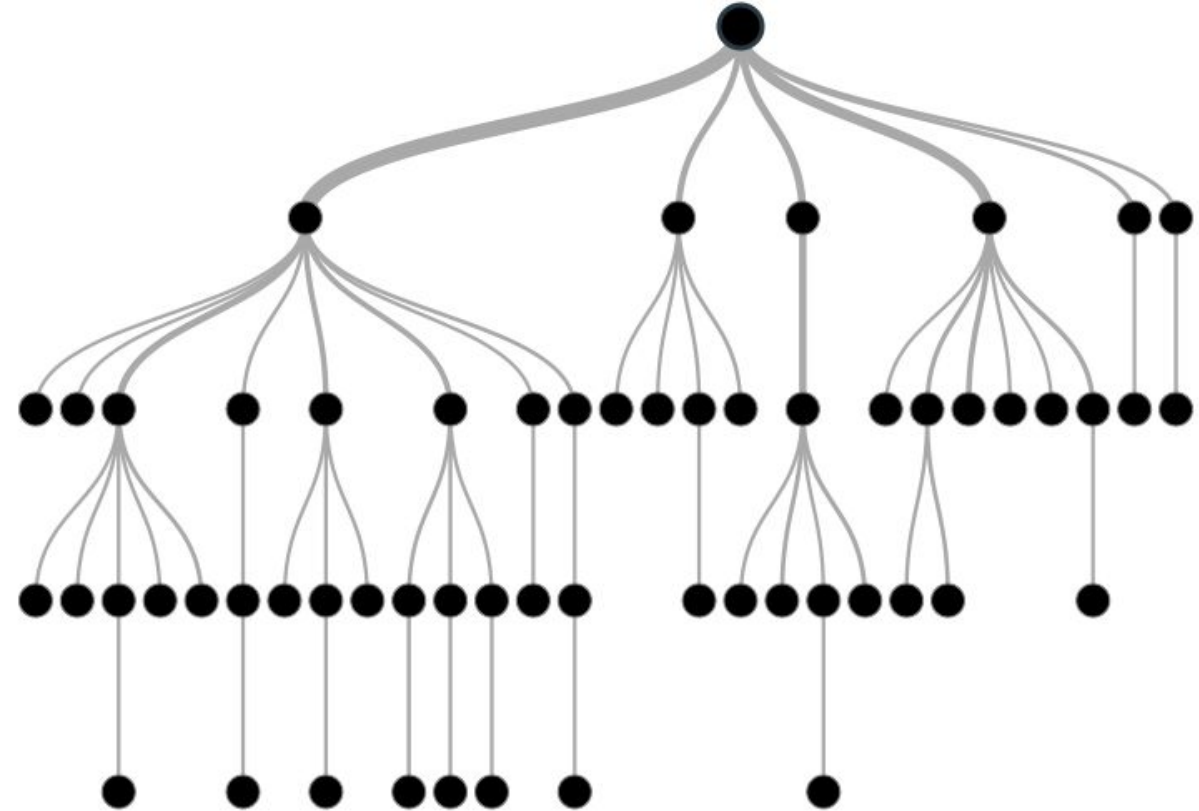
División de un nodo



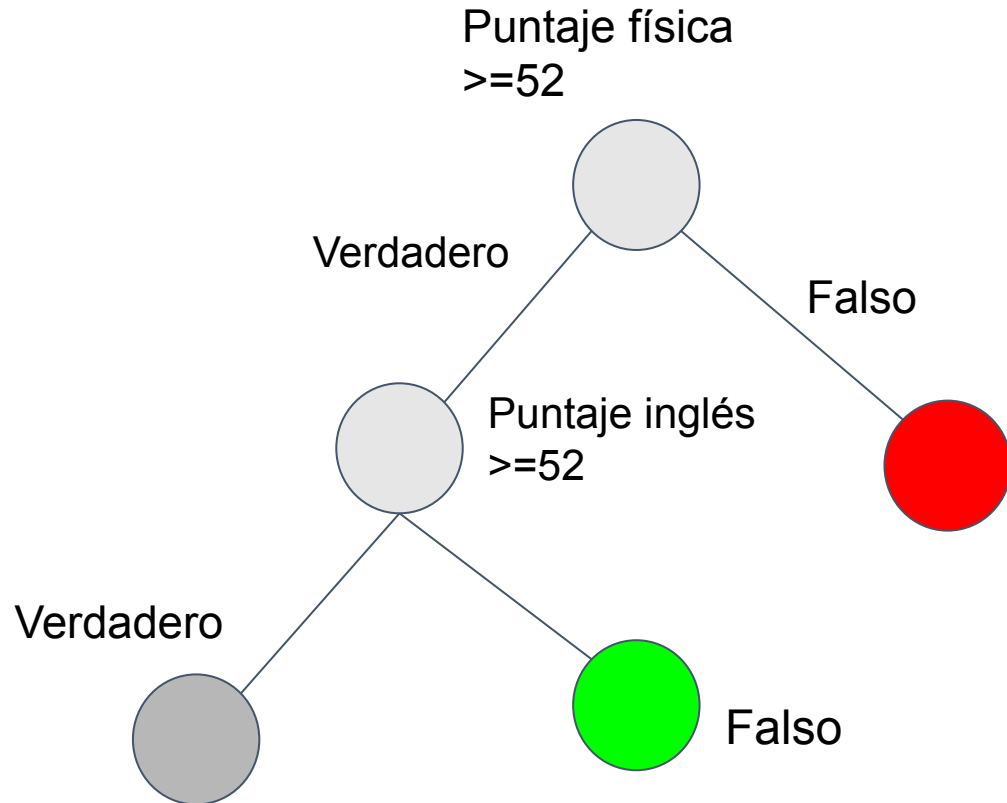
Los nodos serán divididos por medio del algoritmo ID3 y usando la impureza de gini, con una condición que cuenta con una variable, un operador y un valor a comparar

	Complejidad en tiempo	Complejidad en memoria
Modelo de entrenamiento	$O(N^2 + 2^m + k)$	$O(N^2 + M^2)$
Modelo de prueba	$O(N^2 + 2^m)$	$O(n * m)$

N es el número de ejemplos, M es el número de condiciones y k es la constante del método de hash



Modelo del árbol de decisión



Un árbol de decisión binario para predecir los resultados de Saber Pro basado en los resultados de Saber 11

Características más relevantes



Física

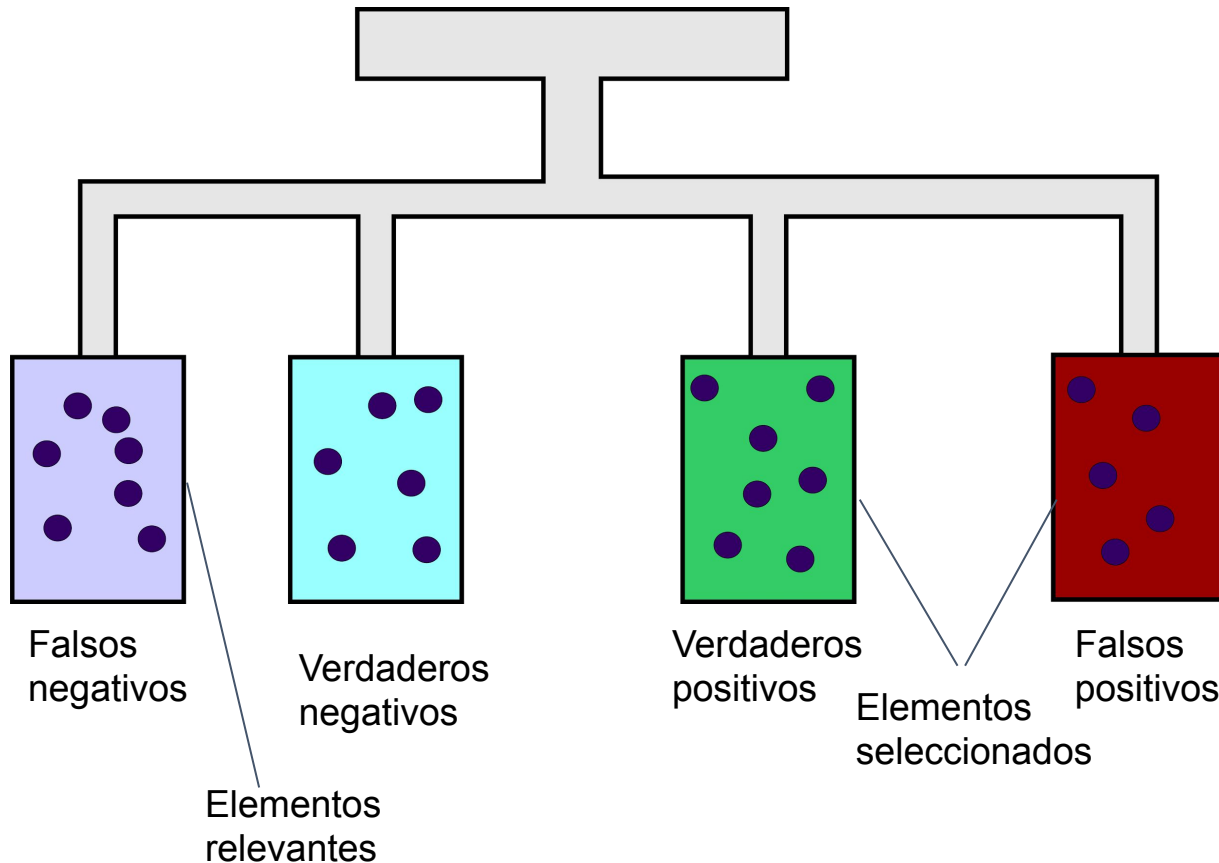


Inglés



Filosofía

Métricas de evaluación



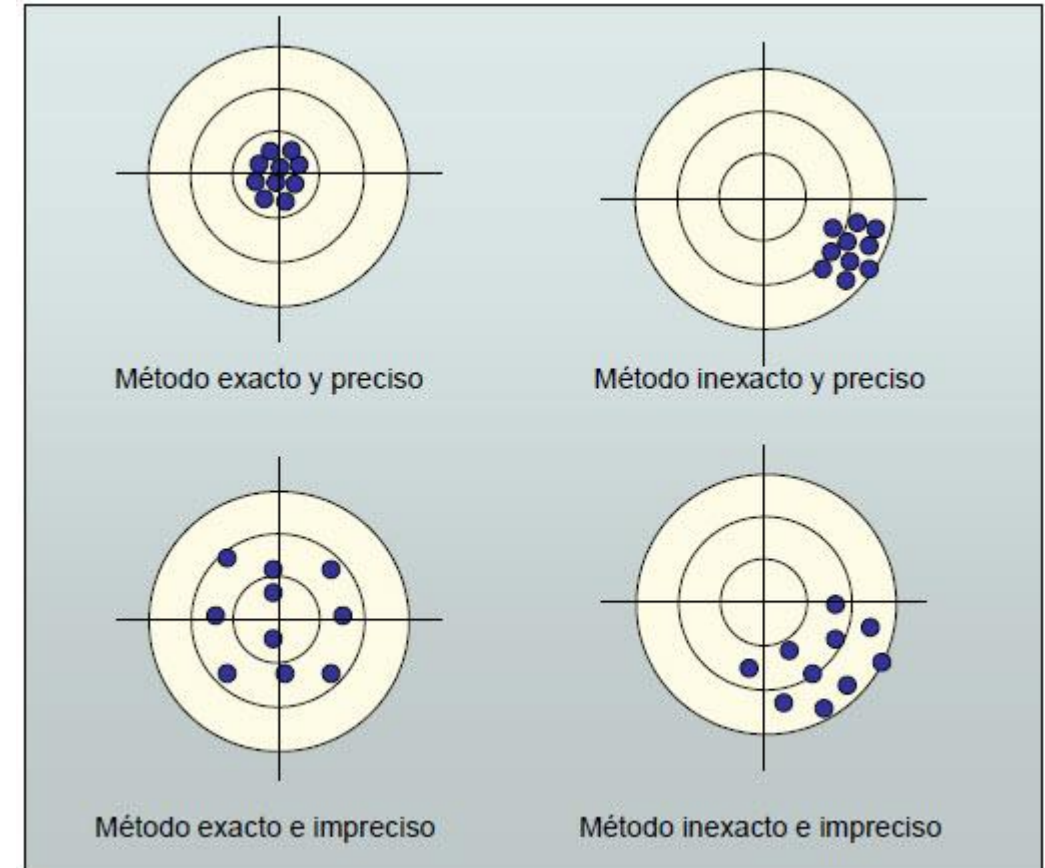
$$\text{Exactitud} = \frac{\text{Pos verdadero} + \text{neg verdadero}}{\text{Total}}$$

$$\text{Precisión} = \frac{\text{Positivo verdadero}}{\text{Positivos predichos}}$$

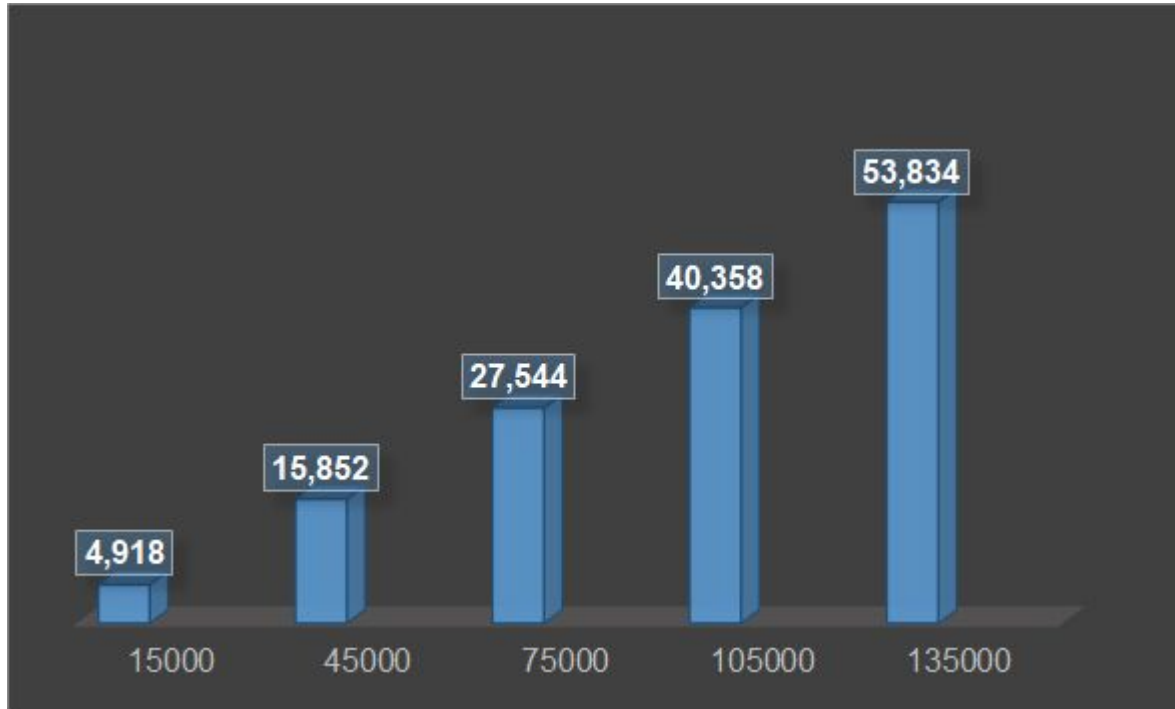
$$\text{Sensibilidad} = \frac{\text{Positivos verdaderos}}{\text{Positivos reales}}$$

	Datos entrenamiento	Datos de prueba
Exactitud	0.8	0.7
Precision	0.8	0.7
Sensibilidad	0.8	0.8

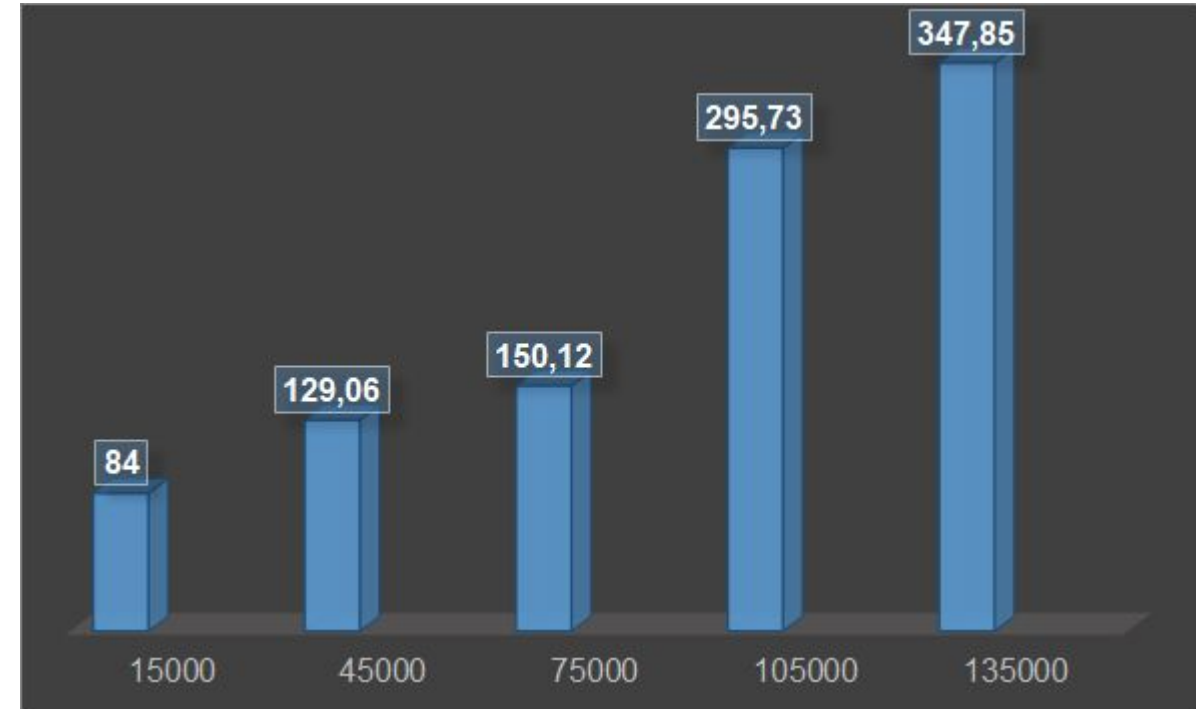
Valores resultantes de usar los datos de 135000 estudiantes en entrenamiento y 35000 para la prueba



Consumo de tiempo y memoria



 Time Consumption



 Memory Consumption

```
*
* @var boolean
*/
define('PSI_INTERNAL_XML', false);

if (version_compare("5.2", PHP_VERSION, ">")) {
    die("PHP 5.2 or greater is required!!!");
}
if (!extension_loaded("pcre")) {
    die("phpSysInfo requires the pcre extension to php in order to work properly.");
}

require_once APP_ROOT.'/includes/autoload.inc.php';

// Load configuration
require_once APP_ROOT.'/config.php';

if (!defined('PSI_CONFIG_FILE') || !defined('PSI_DEBUG')) {
    $tpl = new Template("/templates/html/error_config.html");
    echo $tpl->fetch();
    die();
}
```

GRACIAS